



CENTRE DE RENNES

IRISA

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
BP 105
78153 Le Chesnay Cedex
France
Tél 954 90 20

Rapports de Recherche

N° 198

**ESTIMATION DE MOUVEMENT
DANS UNE SÉQUENCE
D'IMAGES DE TÉLÉVISION
EN VUE D'UN CODAGE AVEC
COMPENSATION DE MOUVEMENT**

Claude LABIT

Mars 1983

Campus Universitaire de Beaulieu
Avenue du Général Leclerc
35042 - RENNES CÉDEX
FRANCE
Tél. : (99) 35.20.00
Télex : UNIRISA 95 0473 F

ESTIMATION DE MOUVEMENT DANS UNE SEQUENCE D'IMAGES DE TELEVISION EN VUE D'UN CODAGE AVEC COMPENSATION DE MOUVEMENT

Claude LABIT

RESUME : Le but de l'étude est la construction d'un estimateur de mouvement point à point adapté à une séquence d'images de télévision. Cet estimateur doit pouvoir tenir compte de la nature quelconque de la scène et des brusques transitions de mouvement. Pour ce faire, nous estimons une information-mouvement a priori sur les points-contours avec l'hypothèse de base sous-jacente que les brusques ruptures de mouvement se situent sur les contours spatiaux d'une image. La propagation point-à-point de l'estimateur utilise des algorithmes adaptatifs avec détecteurs de rupture sans retard. Plusieurs algorithmes ont été testés et comparés sur deux séquences d'image (type visiophone et type télévision) à caractéristiques différentes.

ABSTRACT : An approach for studying pel-by-pel motion-estimator fitted to a sequence of television pictures, is presented. This estimator has to take into account abrupt jumps in motion which can occur within a picture. Thus, we assume that these motion-jumps are located along boundaries of objects ; so a specific edge motion estimation is needed. We include this edge-motion estimator into a complete pel-by-pel motion estimator which combines adaptive propagation and instantaneous detection of motion breaks. Some various algorithms are tested and compared on two quite different picture-sequences.

REMERCIEMENTS

Ce travail a été réalisé au laboratoire d'Automatique de l'IRISA en liaison avec le CCETT (*).

Avant de le présenter, je voudrais d'abord exprimer toute ma reconnaissance et mes remerciements à tous ceux qui, d'une manière ou d'une autre, m'ont aidé à mener à bien ce travail.

J'adresse tous mes remerciements à J.P. VERJUS, *Directeur de l'IRISA* qui a bien voulu m'accueillir dans son laboratoire et me fait l'honneur de présider ce jury.

Je remercie vivement A. BENVENISTE, *Directeur Adjoint de l'IRISA et Responsable du Laboratoire d'Automatique* avec lequel j'ai principalement travaillé. Je tiens à lui dire combien j'ai apprécié son aide et ses avis et sa disponibilité tout au long de l'étude.

Je tiens également à exprimer mes remerciements :

à F. KRETZ, *Ingénieur au CCETT* et à tous les *Chercheurs* de son équipe pour leur aide et les moyens qu'ils ont mis à ma disposition ;

à G. MICHEL, *Ingénieur au CNET* et aux *Chercheurs de l'Atelier micro* pour m'avoir permis de regouter aux joies de l'informatique cablée ;

à J.F. ABRAMATIC, *Ingénieur de recherche à l'INRIA* et Monsieur CASTAN, *Professeur à l'Université P. Sabatier de Toulouse* pour l'intérêt qu'ils ont bien voulu porter à ce travail en participant à ce jury ;

à Madame E. LEBRET pour son aide amicale et efficace ;

à tous ceux qui ont su créer au Laboratoire d'Automatique une ambiance jeune, souriante, sympathique, sportive, etc... et stimulante ;

J'adresse enfin toutes mes félicitations à Huguette BECHU pour avoir mené avec compétence le travail dactylographique.

(*) - Centre Commun d'Etudes de Télédiffusion et Télécommunications.



A ma famille et mes amis

On pourrait mener sans encombres
une belle histoire
d' amour,
en gommant toutes les ombres
en ne gardant que les contours;
Je vous dirais, laissez vous faire
J'ai tout compris, c'est comme ça,
Sans dire que je désespère
De n'y rien comprendre parfois.

(ANNE SYLVESTRE)

S O M M A I R E

Résumé Introductif	Page 1
1) généralités	Page 2
2) cadre de l'étude	Page 3
3) plan détaillé	Page 4
4) données utilisées	Page 7
1ère PARTIE : Position du problème, objectifs et contraintes	
1) Résumé bibliographique	Page 11
1.1) aspect reconnaissance des formes	Page 12
1.2) aspect codage interimage	Page 14
a) systèmes à rafraichissement conditionnel.	Page 14
b) systèmes à compensation de mouvement.....	Page 16
c) conclusions	Page 21
2) Problèmes critiques en codage T.V.....	Page 22
3) Approche adoptée	Page 24
3.a) but et contexte de la thèse	Page 24
3.b) Information à priori sur les ruptures de mou- vement	Page 27
3.c) Présentation des deux autres chapitre	Page 29
2ème PARTIE : Détermination du mouvement des contours	
1) Traitement des données	Page 33
1.1) séquence d'images	Page 33
1.2) définition et détections d'éléments de contours en mode spatial	Page 36
1.3) définition de contours spatio-temporel	Page 39
2) Algorithme de suivis de contours	Page 43
2.1) présentation de l'algorithme	Page 43
2.2) résultats	Page 49

2.3) extensions possibles	Page 53
Processus de validation de contours	Page 54
Reconstruction des contours validés	Page 58
2.6) Résultats et conclusions	Page 59
3) Estimation de mouvement de contours : algorithmes MC1 et MC1 bis	Page 63
3.1) application de l'algorithme	Page 63
3.2) modifications dues au mode temporel	Page 64
3.3) résultats	Page 66
Conclusions sur les algorithmes MC1 et MC1 bis	Page 67
4) Estimateur "simplifié" : algorithme MC2	Page 72
4.1) observation locale du mouvement	Page 72
4.2) Modification de l'état de l'automate	Page 74
Résultats	Page 75
5) Tableau récapitulatif, sommaire de la méthode MC2.....	Page 80
3ème PARTIE : Estimation du mouvement en tout point d'image	
1) Objectifs, type d'algorithmes visés.....	Page 85
2) Comparaison de diverses propagations sans rupture	Page 86
2.1) types de propagations choisies	Page 86
2.2) résultats	Page 87
3) Propagation avec rupture de mouvement	Page 88
3.1) réinitialisation systématique de l'estimateur....	Page 88
3.2) tests de rupture	Page 89
résultats	Page 91
remarques sur la quantification et les interpo- lations	Page 92
Conclusions, problèmes demeurants	Page 111
Bibliographie	Page 113
Annexe I	Page 117
Annexe II	Page 119
Annexe III : Résultats photographiques.....	Page 121

RESUME INTRODUCTIF

Le volume d'informations de toute nature (téléphone, images, documents écrits, données diverses) produites et diffusées quotidiennement est un facteur en forte expansion ces dernières années et correspond à des besoins et des motivations fort divers mais qui tous, ont pour conséquence la mise en place de nouveaux systèmes de communication, de transmission et d'archivage. Afin d'utiliser au mieux les moyens actuels de transmission (câbles, satellite,...) pour absorber ce volume croissant de communications, de nombreuses recherches sont entreprises, ces dix dernières années, avec comme objectifs principaux d'étudier des procédés de compression d'informations.

L'idée de base commune à l'ensemble de ces "traitements de l'information" est d'extraire d'une source d'information la partie "utile" et non redondante de l'information afin de ne transmettre, visualiser ou archiver que celle-ci. Les techniques permettent d'assurer un gain en complexité pour les systèmes de communication (gain en débit) ou d'archivage (gain en volume de stockage).

Cependant la nature tout à fait subjective de la notion d'information "utile" montre bien sa dépendance avec l'application visée : il s'agira tantôt d'une extraction d'objet, d'une paramétrisation, tantôt d'un codage de l'information.

En ce qui concerne la transmission d'images, le but du codage est de supprimer les redondances d'information afin de diminuer le débit global binaire à transmettre.

Notons à ce propos qu'un signal de télévision noir et blanc codé sur 8 bits conduit à un débit numérique de 96 Mb/s si l'on échantillonne le signal vidéo à la fréquence de Shannon, à un débit de 108Mb/s si l'on échantillonne à la fréquence d'échantillonnage normalisée actuelle (Janv.81 = 13.5Mhz)

Les organisations européennes désirent pouvoir ramener ce débit de transmission à un débit normalisé compatible avec les autres transmissions numériques. Le protocole de transmission retenu pour l'exploitation numérique des signaux couleurs de télévision est une transmission

- en composantes séparées
- à un débit de 34.368 Mb/s

Cet objectif montre l'importance du taux de compression qu'il faut atteindre, notamment sur le signal le plus "porteur d'information", le signal de luminance.

L'introduction de nouvelles technologies (fibres optiques...) pour la transmission rend moins critique ce problème de gain en débit mais ne résoud en rien celui d'absorber un volume toujours croissant d'informations. De plus les notions d'extraction de "l'information utile" et suppression des redondances sont générales et peuvent être appliquées à de nombreuses autres applications que la transmission, par exemple l'archivage.

Pour lever toute ambiguïté sur le terme "codage" employé ici, nous distinguerons dans le schéma classique d'un système de communication numérique (cf Fig. R1) au niveau de l'émetteur deux types de codeurs :

- le *codeur-source* qui a pour but d'extraire l'information utile du signal source
- le *codeur-canal* qui, lui, code (avec un code à débit minimal mais peut-être légèrement redondant pour être robuste aux erreurs de transmission) cette information utile et l'injecte dans le canal.

Au récepteur la même distinction peut être faite.

Le terme de codage employé dans toute cette étude ne concerne que le premier type de codeur.

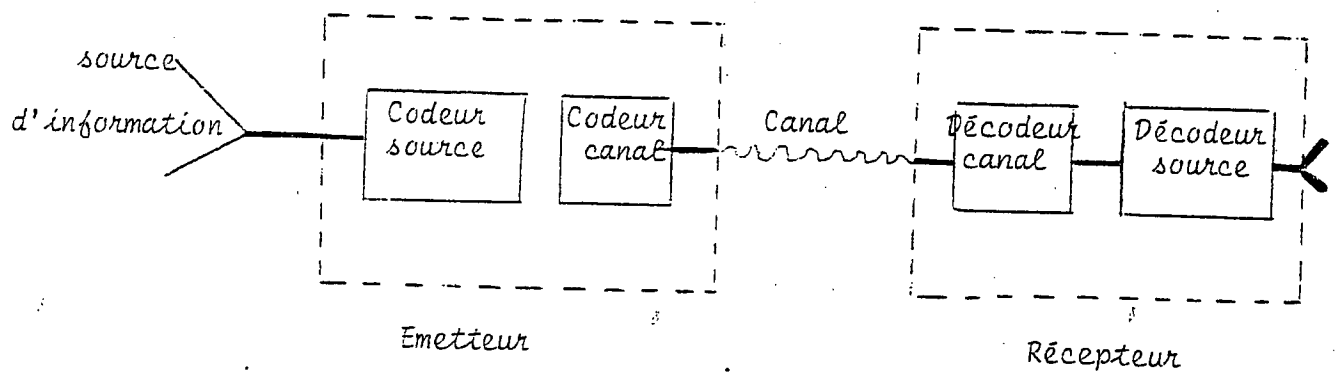


Fig. R1

I - GENERALITES

De nombreuses études ont déjà été menées sur le codage MICD (modulation par impulsion et codage différentiel) qui, au lieu de transmettre toute l'information, ne code que des signaux différentiels d'erreurs entre le signal source et le signal de prédiction. Le schéma de prédiction choisi dépend de la nature de la redondance d'information que l'on désire supprimer.

Dans une séquence d'images, cette redondance est perceptible à deux niveaux :

- à l'intérieur d'une même image
- d'image à image (ou de trame à trame)

Historiquement les travaux sur le codage d'images ont d'abord porté sur le *codage intra-image* c'est à dire à l'intérieur d'une même image. La nature de la redondance que l'on cherche à extraire est dans ce cas purement spatiale. Plusieurs méthodes ont été comparées, notamment entre des prédicteurs monodimensionnels et bidimensionnels. L'introduction de la notion de codage MICD adaptatif (= ADPCM) a permis d'affiner le schéma de prédiction et de réduire en conséquence le débit de transmission d'informations ; dans ce cadre, le choix du prédicteur d'un point d'image dépend de la nature de ce point ; on distingue par exemple les points appartenant à des zones-contours et les points de zones homogènes.

La redondance d'image à image ou *redondance temporelle* concerne :

- *le codage interimage* ; elle est fonction du taux de mouvement ou de changement à l'intérieur de la scène animée puisque si ce taux est faible, très peu d'informations nouvelles (donc utiles) existent entre une image et sa suivante et vice-versa ; les deux cas extrêmes étant :
 - . *la scène fixe* où à l'exception de la 1ère image, le reste de la séquence est redondante et inutile au point de vue informationnel.
 - . *le changement de plan de caméra* où par contre n'existe aucune corrélation entre les deux images insérant le changement de plan. Toute l'information est "*utile*" au sens théorie de l'information (et non pas pour l'oeil humain).

Les travaux déjà menées dans ce domaine ont porté principalement sur des images visiophones à faible taux et variété de mouvements. La redondance temporelle est extraite de l'information utile par détection et estimation du mouvement dans la séquence, estimation plus ou moins affinée suivant le type de codage interimage envisagé.

II - CADRE DE L'ETUDE

L'étude présentée ici porte sur la conception d'un estimateur de mouvement en vue d'un codage interimage avec compensation de mouvement. L'information-mouvement estimée doit permettre de recalculer temporellement les images et ainsi de mieux prédire la source.

L'esprit dans laquelle elle a été menée est dans un premier temps d'étudier de façon la plus complète possible la nature du mouvement dans une séquence numérique de luminance. D'autre part les algorithmes possibles d'estimation de mouvement que nous avons testés s'inscrivent tous dans l'optique de les utiliser pour un système de codage ce qui suppose le respect des contraintes liées au codage télévision et le souci d'une implantation hardware minimale en complexité. De plus des contraintes de qualité subjective et liées au récepteur visuel humain sont aussi à prendre en compte.

III - PLAN DETAILLE

1ère partie : *Résumé bibliographique et approche adoptée*

* Nous commençons cette étude par une recherche bibliographique sur l'ensemble des travaux concernant le mouvement dans les images ; il apparaît que ces travaux concernent aussi bien le domaine de la reconnaissance des formes (extraction et reconnaissance d'objets en mouvement recalage d'images, archivage de paramètres temporels etc...) que le domaine du codage d'images qui vise à une réduction du débit de transmission.

Les contraintes de débit, de qualité et de temps réel sont fortement différents cependant selon le domaine envisagé.

Il convient de souligner que dans le codage intérimage, les études d'algorithmes ont porté pour le plupart sur des images à "normes visio-phoniques" et non pas sur des images télévision.

L'application au domaine de la télévision pose en effet des difficultés spécifiques à celui-ci du fait de l'absence de toute information a priori sur le contenu des images ainsi que sur la nature des mouvements, l'estimateur de mouvement à construire doit pouvoir en prendre en compte n'importe quel type de scène (plusieurs objets, recouvrement d'objets) ou n'importe quel type de mouvement (plusieurs amplitudes de mouvement différents, mouvements véritables ou déformations etc...)

* L'estimation de mouvement est faite dans une optique "traitement de signal", par calcul d'un paramètre temporel qui sera le reflet d'une variation temporelle de la luminance ; la véritable cause (mouvement, variation d'illumination, recouvrement d'objets...) et la nature du mouvement s'il existe (translation, rotation, déformation...) importe peu.

* Comme hypothèse de base nous choisirons de localiser les *transitions brutales* de mouvement sur les "frontières" d'objets. Cela suppose préalablement une détection des contours spatiaux et ensuite un suivi de leur mouvement.

* Dans le but d'une minimisation des informations à mémoriser, nous avons choisi de modéliser les contours d'une image par des *automates d'états*. Le "passé" d'un contour est ainsi résumé dans un vecteur d'état. Ce vecteur d'état est réactualisé ligne à ligne (ou image à image suivant le type de lissage envisagé) le long du contour par des algorithmes séquentiels de filtrage.

* Dans l'image entière, l'estimation et la propagation du mouvement se fait point à point par des *algorithmes adaptatifs* déjà testés dans le cadre d'images visiophoniques. Lors du franchissement d'une transition brutale de mouvement, la propagation adaptative peut être rompue et dans ce cas l'estimateur est réinitialisé en fonction de l'information à priori définie précédemment.

En chaque point d'image, nous disposerons donc d'une information très locale sur le mouvement présent, information dont nous nous servirons pour la définition d'un schéma de codage avec compensation de mouvement.

2ème partie : Estimation du mouvement des contours

L'originalité du travail repose sur la définition de trois approches possibles pour l'estimation de mouvement sur les contours.

* Tout d'abord est fait un rappel sur la méthode et les notations employées dans l'algorithme de détection et suivi de contours spatiaux développé dans la thèse de C. RICHARD. La présentation adoptée est celle par automates d'état associées à chaque contour spatial. Cet algorithme peut être modifié par simple extension de l'automate. Le traitement spatial des contours devra toujours être effectué simultanément aux algorithmes eux-mêmes d'estimation de mouvement.

Les algorithmes d'estimation de mouvement sont :

. *algorithme MC1* : Cet algorithme vise à exploiter la redondance temporelle du mouvement qui se situe sur une même ligne d'image ; le mouvement détecté sur la n ème ligne d'un contour de l'image est en général fortement corrélé avec le mouvement présent à la n ème ligne du même contour à l'image suivante ($t+1$)

. *algorithme MC1bis* : Une deuxième redondance (de nature spatiale cette fois) en ce qui concerne le mouvement existe également ; il s'agit d'utiliser la corrélation existante entre les mouvements présents dans deux lignes consécutives d'un contour spatial d'une image. Ces deux hypothèses de redondance permettent d'effectuer un double lissage (spatial+temporel) sur l'ensemble des observations mouvement.

algorithme MC2 : Cet algorithme n'utilise que la seule redondance spatiale décrite précédemment et cela sans perte notable de performances pour l'estimateur.

Ces trois algorithmes sont basés sur la description des contours d'images par automates d'état. La comparaison en complexité portera donc sur le nombre d'automates à gérer globalement pour l'estimation du mouvement des contours d'une image.

3ème partie : Estimateur de mouvement en tout point

* L'estimateur de mouvement en tout point de l'image repose sur la technique classique des algorithmes d'estimation adaptatifs minimisant une fonctionnelle d'erreur de prédiction. Le type de prédiction utilisé est la prédiction avec compensation de mouvement au sens de la figure R2 :

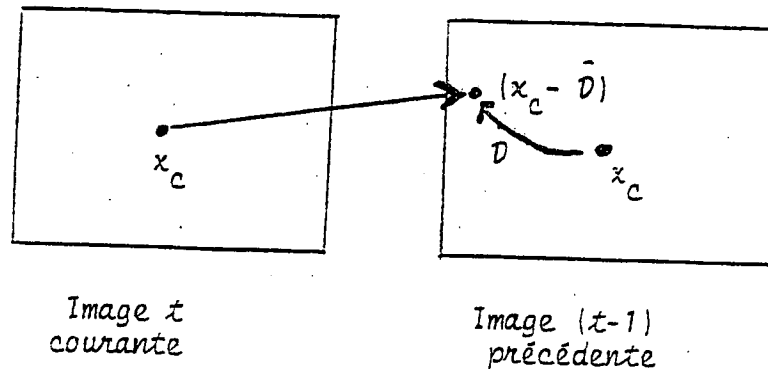


Fig. R2

le point courant x_c d'intensité $I(x_c, t)$ est prédit par le point $x_c - \hat{D}$ où \hat{D} est l'estimation de mouvement du point x_c

L'erreur de prédiction à minimiser est donc :

$$e = I(x_c, t) - I(x_c - \hat{D}, t-1)$$

Plusieurs travaux (notamment ceux du Dr NETRAVALI) ont déjà été menés sur ce sujet, nous rappelons et testons divers choix de fonctionnelle d'erreur dans cette troisième partie.

* Cependant ce type d'algorithmes d'estimation ne peut prendre en compte sans retard en convergence, les ruptures brutales de mouvement. Il convient donc d'ajouter à l'estimateur adaptatif un détecteur de ruptures brusques de mouvement ; grâce à l'hypothèse faite sur la position des transitions brutales de mouvement, les détecteurs cherchés sont fournis par l'algorithme d'estimation du mouvement des contours présenté au chapitre précédent.

Puisque l'optique de cette étude est l'application de l'estimateur construit pour un schéma de codage intérimage, nous présentons les résultats comparatifs des diverses méthodes d'estimation sous forme d'images et d'histogrammes d'erreurs de prédiction.

IV - DONNEES UTILISEES

Les simulations faites sur l'IRIS 80 du CCETT^(*) (Rennes) ont utilisé deux types de séquence :

- la séquence "*visiophone*" *Judith* : 50 images format (256x256) dans lesquelles le mouvement ne dépasse que très rarement en amplitude 1 à 2 pels/image.

- la séquence "*télévision*" *Chateau* : 50 images format TV (624x576) le mouvement peut y être beaucoup plus rapide (5 pels/image) mais dépend fortement de la position de l'image dans la séquence.

La plupart du mouvement provient en grande partie de la caméra.

Pour des raisons de volume mémoire disponible, de nombreux résultats présentés portent sur des zones caractéristiques (zone homogène, zone contour , zone texture) extraites d'une image et suivies temporellement. Ceci n'influe pas sur la construction elle même de l'algorithme d'estimation du mouvement.

(*) - Centre commun d'études de télédiffusion et télécommunications.

1ere PARTIE

POSITION DU PROBLEME OBJECTIFS ET CONTRAINTES

I - RESUME BIBLIOGRAPHIQUE

Analyse des principales études sur le mouvement dans les images.

1-0 :

L'étude du mouvement dans une séquence d'images donnée peut prendre des formes très diverses, d'une part dues à la nature elle-même de la séquence, d'autre part à l'application et l'utilisation de cette information-mouvement.

Qu'il s'agisse d'une séquence d'images-visiophone, de télévision, d'une série de prise de vues aériennes, astronautiques ou médicales, la gamme de problèmes et d'objectifs à résoudre est très étendue. Ces applications diverses se différencient par :

- *nature de l'image* :
 - . type d'échantillonnage
 - . qualité à assurer
 - . information utile
- *nature de la séquence*:
 - . type d'échantillonnage temporel
 - . nature du mouvement
 - . taux de mouvement (critère de mesure à définir)
- *information mouvement*:
 - . estimation locale ou globale du mouvement dans l'image
 - . précision nécessaire de l'estimation
 - . utilisation de cette information de mouvement

L'étude exhaustive de l'ensemble des applications visées est faite par H-H Nagel (1) avec pour chacune d'elles les techniques spécifiques de traitement qui lui sont associées. Elles concernent le codage d'images et la compression de débit de transmission de ces images, les images satellite (observations météorologiques répétées par exemple), le domaine médical (tomographie temporelle) aussi bien que la robotique (saisie de pièces en mouvement), le contrôle de systèmes industriels et le contrôle du trafic routier... Toutes ces applications ont en commun l'objectif de considérer le temps comme une quatrième composante utile et de rechercher les similitudes (ou l'absence de similitudes) d'une image à l'autre de la séquence.

Une séquence d'images se caractérise également par la nature du mouvement que l'on y trouve. Une liste des divers types de mouvement (que l'on rencontre en télévision mais cette typographie présente un caractère général) est faite dans (2) avec le souci de montrer quelles sont les dégradations en télévision numérique que ces mouvements occasionnent.

Pour notre part, nous classifions les travaux menés sur le mouvement en deux familles :

- la première s'inscrit plus dans un aspect "*Reconnaissance des formes*" du traitement d'images pour lequel la détermination du mouvement est un objectif en soi.
- la seconde concerne l'aspect "*codage d'images*" dans lequel l'estimation du mouvement sera utilisé dans un schéma prédiction pour mieux prédire l'image courante. Il s'agit dans ce cas de supprimer la redondance temporelle inutile qui existe à l'intérieur d'une séquence.

1-1 : - Aspect "*Reconnaissance des formes*"

L'idée de base de cette approche est d'associer aux objets détectés et classifiés dans une image (par un algorithme classique de reconnaissance de formes) un vecteur déplacement qui ajoute donc une information sur l'évolution temporelle de l'image.

Les objets peuvent être connus (information à priori sur le type de scènes) tout comme la nature du mouvement de ceux-ci (uniquement des mouvements de translation, ou de rotation...)

Les informations à priori permettent des techniques plus simples et plus rapides pour la détection et l'estimation du mouvement. Elles perdent cependant leur caractère général par le modèle d'objets et de mouvement choisis.

Cette approche se retrouve fréquemment dans la poursuite de cibles, le recalage d'images météorologiques ou médicales, le contrôle du trafic routier [3] , [4] , [5] .

Les techniques employées, concernent principalement la segmentation et la classification de l'image en zones fixes ou zones en mouvement, arrière plan ou objet-cible. Bien souvent les contraintes temporelles ne sont pas trop sévères ce qui permet parfois, avant l'étude proprement dite du mouvement, un pré-traitement sur l'image (filtrage, restauration, passage dans les domaines transformées...) qui facilite l'extraction du mouvement et la reconnaissance des objets.

L'estimateur de mouvement est de façon générale, construit de manière globale puisque, avec cette approche "*Reconnaissance de formes en mouvement*", l'idée est d'associer à chaque objet détecté ou à chaque partie segmentée de l'image un vecteur déplacement. Le calcul de l'information mouvement se fait donc sur un bloc (bien souvent non causal) de points d'images et non pas point à point.

Information

à priori

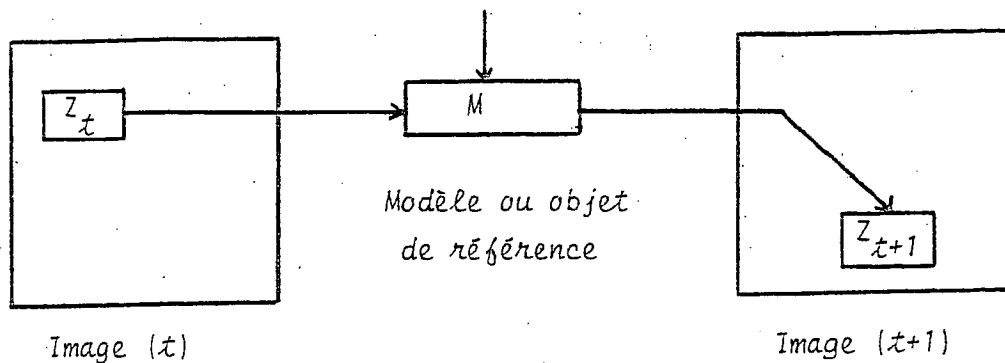


Figure : 1.1.a.

La Figure 1.1a montre le principe de recherche de la nouvelle position d'un objet qu'on a préalablement extrait de l'image (t) ou dont on dispose des informations (t+1) ayant la plus forte corrélation avec la zone Z_t de l'image (t). Un critère de décision permettra de déclarer si Z_{t+1} constitue la nouvelle position de l'objet M. L'estimation du mouvement de M découle alors de façon simple, de la comparaison des deux emplacements respectifs de Z_t dans l'image (t) et de Z_{t+1} dans l'image (t+1).

Réciproquement un segment de l'image courante (t+1) peut se voir attribuer un vecteur déplacement après calcul des diverses corrélations entre ce segment et différents segments de l'image précédente (t) et observation du maximum de cette fonction.

Cette démarche inverse est celle qui est utilisée le plus couramment car l'image (t) étant mémorisée dans une mémoire d'image, l'estimation du mouvement se fait, segment après segment d'images, séquentiellement suivant l'ordre d'arrivée des données. De plus, dans ce cas, le mouvement est calculé à partir des nouvelles observations et non pas à partir de modèle d'objets qui peut-être n'existe plus dans la nouvelle image. La Figure 1.1b résume le principe de ce corrélateur.

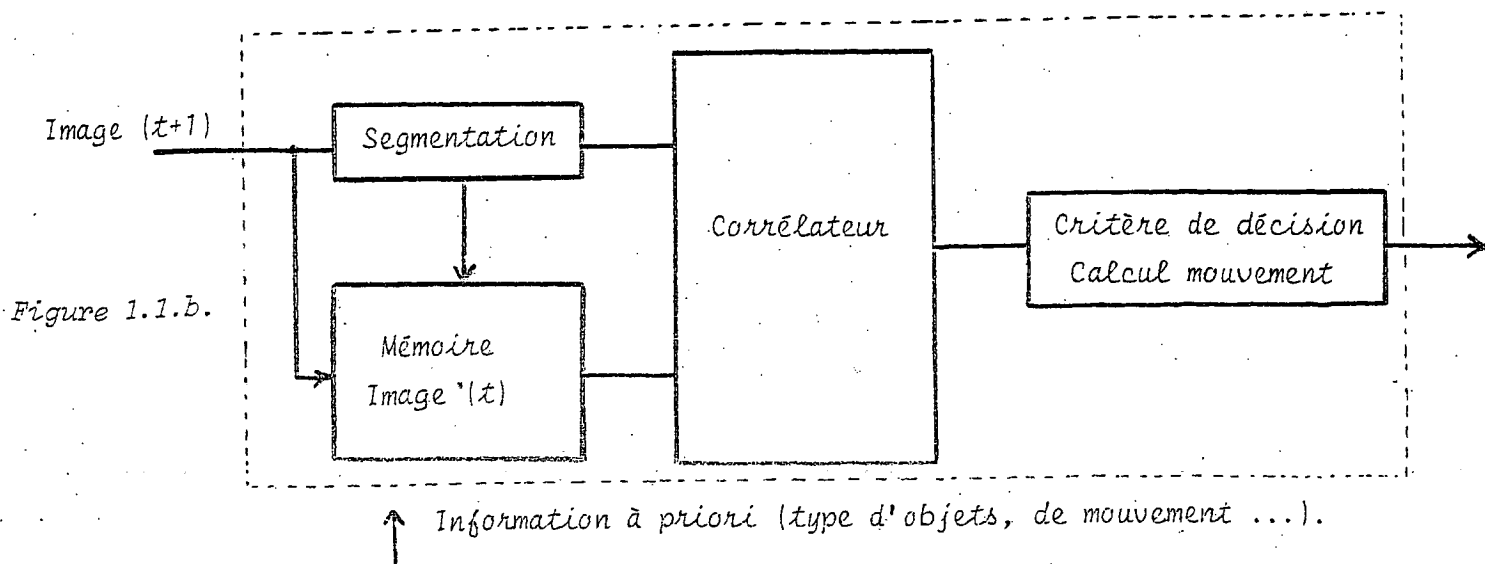


Figure 1.1.b.

Conclusion :

L'aspect "*Reconnaissance des Formes*" considère le calcul du mouvement comme une fin en soi et cette grandeur possède une certaine réalité physique puisqu'il s'agit du recalage de deux objets "*reconnus*" et fortement corrélés entre-eux. Cependant la méthode présente des inconvénients :

- elle est fort dépendante du type d'objets à détecter et du type de scène : * *choix du modèle de référence ?*
* *quelle segmentation ?*

- la nature du mouvement influe fortement sur la taille d'exploration du corrélateur,

- elle met en oeuvre généralement des calculs complexes (d'autant plus que la précision, dans le recalage, désirée est grande) et cela devient lourd si l'on s'impose un fonctionnement *temps-réel*.

Il semble donc que cette démarche ne peut être choisie lorsqu'on ne possède aucune information à priori sur la séquence d'images (comme par exemple en télévision).

1-2 : - Aspect "*Codage interimage*"

De nombreux travaux ont déjà été menés sur le codage interimage avec pour objectif commun la réduction de débit de transmission du signal d'images par utilisation de la redondance temporelle entre deux images ou deux trames successives

L'ensemble de ces études se répartissent en deux familles :

- *les systèmes à rafraichissement fixe ou conditionnel*
- *les systèmes à compensation de mouvement.*

Une analyse et une bibliographie sur les systèmes à rafraichissement se trouvent dans [1] et en annexe de [6]. Nous nous intéressons plus principalement aux systèmes à compensation de mouvement, principe de base de notre étude.

1.2a : *les systèmes à rafraichissement conditionnel [7] à [12]*

* L'idée la plus simple de codeur temporel consiste à transmettre systématiquement sur le canal les erreurs point-à-point quantifiées d'une image à l'autre : il s'agit du *système à rafraichissement fixe*. La Figure 1.2a montre le schéma de ce codec différentiel interimage.

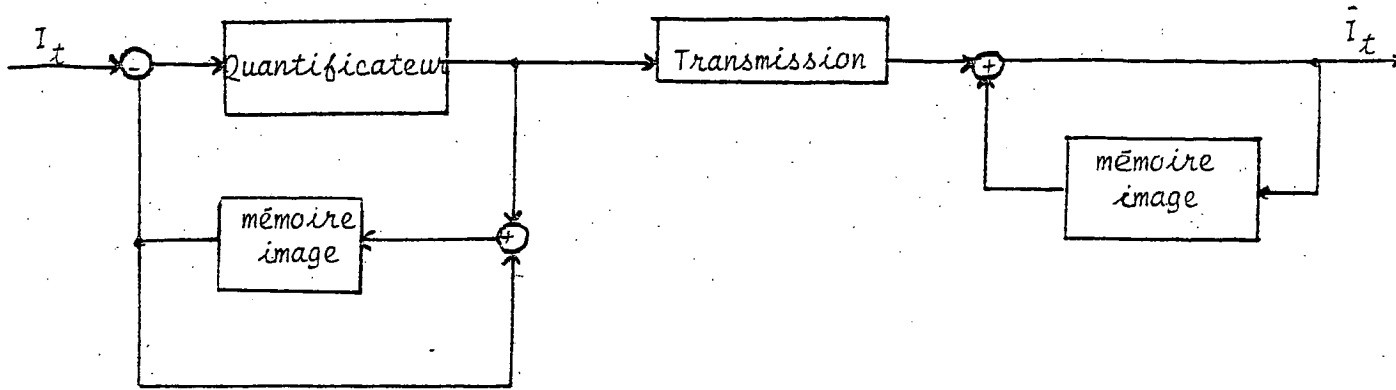


Figure 1.2a

* Le codeur à rafraîchissement fixe n'utilise pas complètement la redondance entre deux images successives puisque même en l'absence d'informations nouvelles, (cas critique du plan fixe) il y a transmission d'informations (erreurs nulles dans ce cas) sur le canal.

L'étape suivante consiste donc à construire un *codeur à rafraîchissement conditionnel* qui distingue deux cas :

- les zones immobiles : aucune information à transmettre
- les zones en mouvement : transmission des erreurs temporelles quantifiées et codées.

Dans ce système de codage, la quantité d'informations à transmettre sur le canal dépend fortement de la *quantité* de mouvement présent dans l'image. Il faut donc, outre le bloc *Détection de mouvement* qui permet de classifier les points (ou blocs de points) d'image en zones fixes ou mobiles, ajouter au schéma 1.2 a une mémoire tampon permettant d'adapter ce débit variable d'informations au débit constant du canal de transmission.

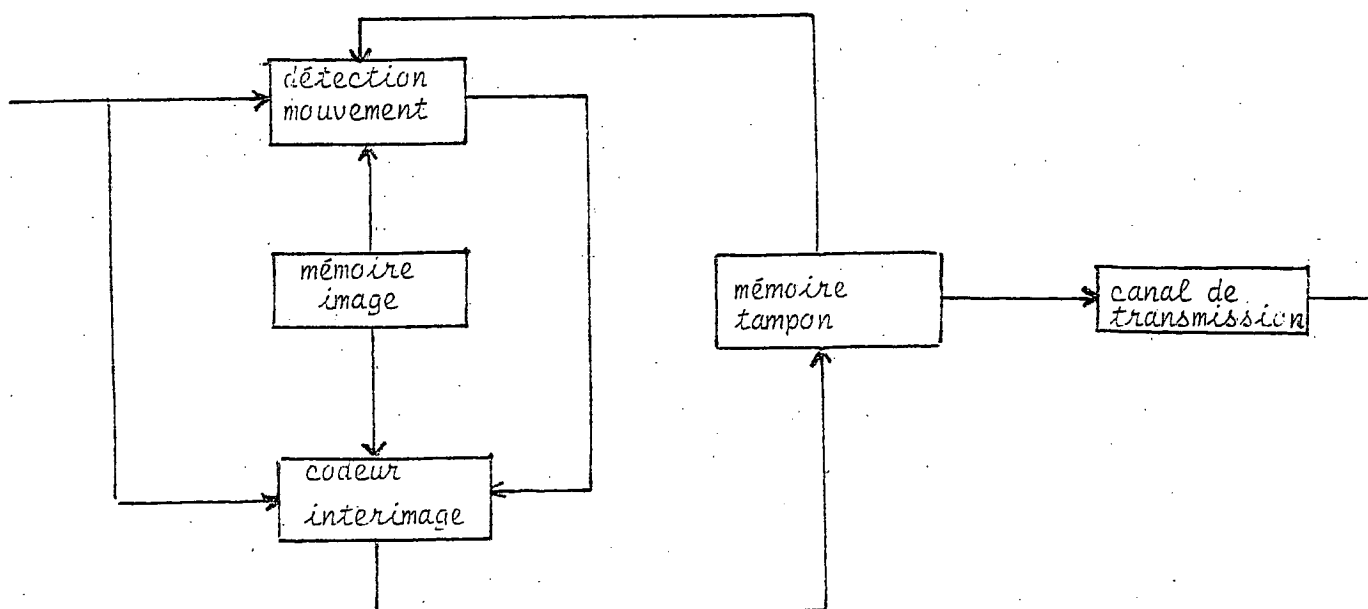


Figure 1.2aa

La complexité de ce codeur réside dans :

- la gestion de la mémoire tampon permettant une utilisation optimale de ce buffer dans les cas extrêmes de mouvement. Les informations sur le remplissage du tampon définissent par une boucle de contrôle les critères de détection de mouvement,

- l'optimisation du codeur : parts respectives prises par les informations d'erreurs et celles d'adressage. Un adressage différentiel peut par exemple être employé 13.

1.2b : - Les systèmes à compensation de mouvement

Ces codeurs utilisent le fait qu'une image dont le mouvement est estimé de manière précise est très bien prédite par l'image précédente translatée. Il convient donc dans le schéma 1.2aa du codeur par rafraichissement conditionnel de doubler le bloc "Détection de mouvement" d'un bloc "Estimation de mouvement".

Cette estimation peut être globale pour toute l'image, par blocs ou point à point. Ce système distingue donc trois types de zones dans l'image :

- (i) - zones fixes : aucune information à transmettre
- (ii) - zones compensables : le mouvement y est "correctement estimé. Aucune information n'est à transmettre également.
- (iii) - zones non-compensables : transmission de l'information brute des erreurs temporelles (ou spatiales).

Dans les trois cas, le type de codage (i), (ii), (iii) est à transmettre.

Notations :

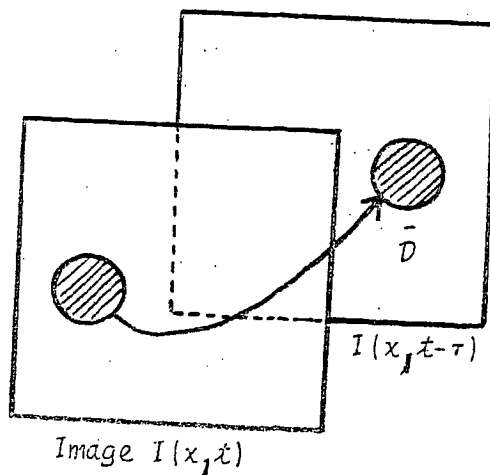


Figure 1.2.b.

$I(x, t)$: intensité du point x à l'image t

$I(x, t-\tau)$: intensité du point x à l'image précédente (τ : échantillonnage temporel)

$D(x, t)$: mouvement réel du point courant

\hat{D} : mouvement estimé

$\vec{\Delta I}(x, t)$: gradient d'intensité

$FD(x, t)$: différences temporelles point à point (Frame difference)

$$FD(x, t) = I(x, t) - I(x, t-\tau)$$

L'hypothèse fondamentale de la compensation de mouvement est :

$$I(x, t) = I(x-D, t-\tau)$$

Remarques :

τ peut être l'intervalle de temps séparant 2 images ou 2 trames. Dans le second cas, si la structure d'échantillonnage est du type quinconce trame [37], l'estimateur de mouvement devra tenir compte du biais apporté dans le calcul des FD .

Nous résumons ici brièvement les études faites sur ce sujet :

* Estimation du mouvement sur un bloc de points : [14] à [19]

Beaucoup d'algorithmes d'estimation du mouvement de translation d'un objet considèrent que cette translation est constante sur un bloc de points d'images.

Limb et Murphy [17] ainsi que C.Cafforio et Rocca [16] proposent par des approches différentes la même expression comme estimateur

$$\vec{\hat{D}} = \left\{ \begin{array}{l} \frac{\sum FD(x) \text{sign } \Delta I_x(x)}{\sum |\Delta I_x(x)|} \\ \frac{\sum FD(x) \text{sign } \Delta I_y(x)}{\sum |\Delta I_y(x)|} \end{array} \right\} \quad (1)$$

La somme porte sur le bloc de points courant inclus dans la même zone en mouvement.

Une extension à trois dimensions est proposée dans [18] en effectuant un lissage temporel de cet estimateur (sur trois trames) ce qui revient à non plus considérer des blocs spatiaux pour l'estimateur mais des blocs de points tri-dimensionnels.

A.N Netravali et IO Robbins [20] apportent une nouvelle modification par estimation différentielle du mouvement :

si DFD ("Displaced Frame Difference") désigne la quantité suivante :

$$DFD(x, \hat{D}^{i-1}) = I(x, t) - I(x - \hat{D}^{i-1}, t - \tau)$$

avec

\hat{D}^{i-1} : valeur de l'estimateur de mouvement à l'itération précédente.

\hat{D}^i : valeur à l'itération courant vaut :

$$\hat{D}^i = \hat{D}^{i-1} - \left\{ \begin{array}{l} \frac{\sum DFD(x, \hat{D}^{i-1}) \text{ sign } \Delta I_x(x)}{\sum |\Delta I_x(x)|} \\ \frac{\sum DFD(x, \hat{D}^{i-1}) \text{ sign } \Delta I_y(x)}{\sum |\Delta I_y(x)|} \end{array} \right\} \quad (2)$$

Il s'agit d'un algorithme de descente où la correction apportée d'une itération à l'autre n'est autre que l'expression différentielle (remplacement des différences temporelles FD par les différences temporelles déplacées DFD) de l'estimateur de Limb et Murphy.

- la somme porte sur le bloc de points inclus dans la zone en mouvement
- le calcul des DFD fait intervenir des valeurs interpolées
- le processus itératif s'arrête lorsqu'un certain critère d'erreurs (portant par exemple sur les DFD) est atteint.

* Estimation point à point du mouvement : [20] à [27]

L'estimateur de mouvement peut être défini au niveau d'un point d'image. L'avantage de cette méthode est de permettre de traiter localement les conflits qui peuvent apparaître dans des scènes à plusieurs objets mobiles.

L'estimateur choisi [20] suit un algorithme adaptatif à gain constant du type :

$$\hat{D}(x_i) = \hat{D}(x_{i-1}) - \epsilon f(\hat{D}_{x_{i-1}}, x_{i-1}, t)$$

La fonction de correction f est calculée afin de minimiser l'erreur,

$E(DFD(x, t)^2)$, l'espérance étant considérée au sens de moyenne prise sur un voisinage du point x

Ceci donne comme algorithme récursif pour l'estimation de D

$$\vec{D}^i = \vec{D}^{i-1} - \epsilon DFD(x_i, \vec{D}^{i-1}) \Delta I(x_i - D, t - \tau) \quad (3)$$

avec comme notations :

$\vec{D}^i = \vec{D}(x_i)$: déplacement du point courant x_i

$\vec{D}^{i-1} = \vec{D}(x_{i-1})$: déplacement du point précédent x_{i-1}

L'itération se fait donc, dans ces études [20] à [27], point à point à l'intérieur de la zone en mouvement. L'estimation se propage point à point, ligne par ligne selon le processus de balayage de la télévision.

L'étude de la convergence de ce type d'estimateur est détaillée dans [28, 29]. Un compromis juste est à trouver entre une convergence rapide (ϵ grand) et un bruit d'agitation résiduelle faible (ϵ petit). La vitesse de convergence se mesure ici en nombre de points d'images.

* Résultats

Les tests de cet estimateur (3) ont été faits sur des séquences d'images artificielles et sur deux séquences d'images réelles de type visiophone ayant respectivement comme pourcentage de points en mouvement 15 à 50% (séquence JUDITH) et 92 à 96% (séquence MIKE et NADINE). Les résultats donnent un gain en débit (= en bit/pel) de l'ordre de 30 à 50% pour la première et 22% pour la seconde par rapport à un système à rafraîchissement conditionnel.

* Remarques

Diverses simplifications peuvent être apportées à (2) sans dégradations notables des performances de l'estimation, du moins sur le type de scènes testées (un objet en mouvement quasi-uniforme). Elles portent sur :

- les interpolations à effectuer dans le calcul des "DFD" et des gradients spatiaux ΔI : interpolation à 4 points à 3 points
troncature entière pour les déplacements intervenant dans ces interpolations ... etc

- la quantification plus ou moins fine des facteurs de correction DFD et ΔI . Le choix du gain en dépend.

- . la formule (3) est calculée sur les valeurs brutes ("quantification" la plus fine) [20.] ($\epsilon = 1/1024$ dans ce cas pour une intensité lumineuse sur 8 bits).

- . la formule (4) donne la quantification la plus grossière :

$$(4) \quad \boxed{\hat{D}^i = \hat{D}^{i-1} - \epsilon \operatorname{sign}(DFD) \operatorname{sign} \vec{\Delta I}} \quad \text{avec} \quad \begin{cases} \operatorname{sign}(x) = 0 \\ \text{pour } x = 0 \\ \operatorname{sign}(x) = \frac{x}{|x|} \\ \text{sinon} \end{cases}$$

* Compléments d'étude sur l'estimateur récursif point à point :

- . l'estimation du mouvement peut aussi être calculé après passage dans un domaine transformé [22,23] (transformations cosinus, Hadamard Haar, slant...). Cela suppose la division de l'image par blocs, le calcul des coefficients de transformation spatiale, l'application de l'algorithme récursif (type forme (3) aux coefficients de transformation.)

Les résultats dépendent fortement de la scène : une comparaison entre le domaine cosinus $2x4$ et le domaine direct Intensité (sur des images Judith) à schéma de prédiction identique montre une amélioration légère de l'ordre de 5 à 10 % en b/pels.

Cependant aucune comparaison de complexité "Hardware" n'a été faite. Le passage dans tout domaine transformé nécessite une complexité de calculs (vectoriels) non négligeable.

Un résultat important toutefois est que les propriétés de convergence de l'estimateur ne dépendent pas du domaine transformé considéré.

- . prise en compte des variations d'illumination de la scène au cours de la séquence [24,25] .

le nouveau modèle d'image choisi dans ce cas est :

$$\begin{array}{ccccc} I(x,t) & = & L(x,t) & \times & R(x,t) \\ \uparrow & & \uparrow & & \uparrow \\ \text{Intensité} & & \text{illumination} & & \text{réflectance} \end{array}$$

Deux compensations distinctes sont menées en parallèle :

- . une compensation de gain
- . une compensation de mouvement.

Quatre systèmes de codage sont comparées sur la séquence *Judith* par référence au système à rafraichissement conditionnel, les codeurs à compensation de gain, à compensation de mouvement, et à compensation mixte gain + mouvement améliorent respectivement en % le débit moyen de 50%, 61% et 63%. Les performances des deux codeurs à compensation de gain et à compensation de mouvement sont inversées lorsque le pseudo-"mouvement" de la scène a pour origine principale une variation de l'illumination.

. fonction de correction pondere [26] ; minimisant la même erreur quadratique $E(DFD^2(x, D))$, l'algorithme adaptatif filtre la fonction de correction ($DFD \times \Delta I$) sur une fenêtre glissante de P points précédant le point courant x_i .

Avec les mêmes notations, la formule (3) devient :

$$\hat{D}_{i-1} = \hat{D}_{i-1} - \epsilon \sum_{y=0}^{p-1} W_j \cdot DFD(x_{i-y}, \hat{D}_{i-1}) \Delta I(x_{i-y} - \hat{D}_i, t_r)$$

Une sensible amélioration des résultats sur cet algorithmes ne se fait vraiment sentir que pour des scènes à mouvement uniforme.

. compensation de mouvement appliqué au signal composite. (luminance + chrominance) [27] ; l'idée de mener trois compensation de mouvement pour chacuné des composantes du signal. L'estimation réelle est alors fonction de ces trois "informations" mouvement. Une estimation vectorielle (afin de de ne pas décorreler les trois composantes) est aussi envisagée.

Les résultats montrent cependant qu'un seul estimateur de mouvement pour les trois composantes est suffisant.

1-2c - Conclusions

Au vu de l'ensemble de ces études, deux remarques s'imposent :

. la plupart des résultats présentés portent sur des séquences de type visioconférence ("head and shoulders") ou dont le contenu du point de vue type et nombre d'objets présents dans la scène, type et nombre de mouvements différents, y ressemble. Or dans une séquence d'images de télévision quelconque, nous ne disposons pas à priori d'informations sur la nature de la scène : de nombreux conflits peuvent apparaitre lors de la présence d'objets à mouvements différents et qui se croisent provoquant ainsi des recouvrements d'objets ou la découverte de parties cachées (sans parler des changements de plan). Il n'existe donc que très peu de littérature et de résultats portant sur de véritables et quelconques séquences d'images de télévision et l'estimation de leur mouvement.

. l'origine du "mouvement" (ou du "changement" comme la nuance est faite dans [19] , est très variée :

- variation d'illumination de la scène
- déplacement réel du point
- bruit

Ces diverses origines peuvent être détectées et distinguées par observation des corrélations spatiales, les différences interimages dues au bruit étant moins corrélées spatialement. Toutefois il est souhaitable d'avoir le même algorithme d'estimation pour l'ensemble des causes possibles de "mouvement" même si le mouvement estimé de cette manière ne correspond pas forcément à un mouvement physique réel. Pour ce faire l'approche locale la plus fine possible (point à point) pour l'estimateur et une adaptation rapide aux changements de mouvements (vitesse de convergence de l'estimateur) sont nécessaires.

II - PROBLEMES CRITIQUES EN CODAGE T.V.

2-1 : - Qualité subjective d'images [30,31,32]

Les dégradations en codage numérique ont plusieurs origines possibles :

- * le système de numérisation : qualité d'échantillonnage (cf Tableau ci-dessous)
- * le système de codage à réduction de redondance
- * les erreurs de transmission.

Le système de codage à construire doit donc permettre une réduction de débit sans perte de qualité subjective pour le système visuel humain et être robuste vis à vis des erreurs de transmission.

Les critères de qualité dépendent du type d'images utilisées : les références citées ci-dessus précisent ceux choisis en codage Télévision ainsi que la typologie des diverses dégradations (flottement de contour, bruit granulaire, bruit de trainage....)

* Tableau voir page suivante

	Nb lignes	Nb lignes utiles	fréquence image (Hz)	Entrelacement trames	bande passante (Mhz)	Nb de points par ligne
Visiophone	271	255	30	2:1	1	256
Standard NTSC	525	485	30	2:1	4.2	422
Standard européen	625	575	25	2:1	6	624

Comparaison de divers signaux d'image

2-2 : - Contraintes (signal T.V.)

i) contrainte de causalité :

Cette contrainte est causée par le processus de balayage T.V. qui se fait séquentiellement point à point (de gauche à droite), ligne par ligne (de haut en bas) et trame à trame. Ce type de balayage conditionne l'allure des masques qu'on emploiera pour les détections et suivi de contours, de mouvements et pour les masques de prédiction : tous ces masques devront être causaux (cf figure 2.a)

Les seules informations utilisables ne peuvent être que celles déjà prédites (à l'émetteur) ou transmises (au récepteur).

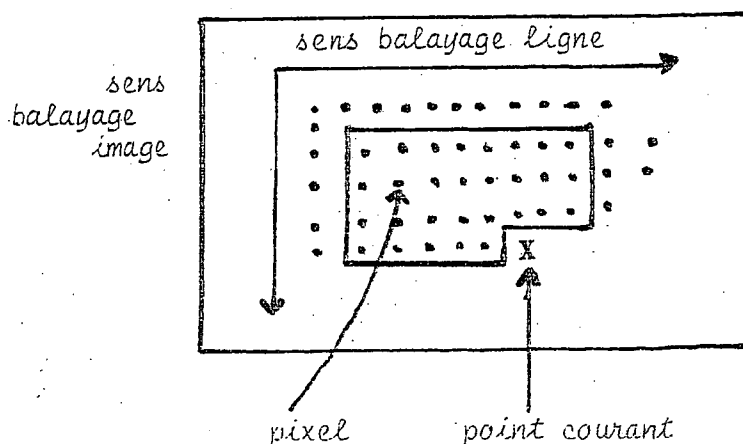


Figure 2.a - Masque causal

Cette contrainte de causalité détruit toute possibilité d'isotropie dans la recherche des contours ou des objets ainsi que de leurs mouvements. L'handicap est lourd comparé aux techniques possibles dans l'approche "*Reconnaissance des formes*" qui elles, ne privilégient aucune direction à priori puisque on peut considérer dans ce cadre l'ensemble de l'image connu à priori.

2-3 : - Contrainte "Temps réel" (cf [35])

La notion de "*Temps réel*" désigne ici le fait que le schéma de prédiction (et tous les calculs dont il a besoin pour fonctionner) doit suivre le flot d'entrée du signal d'image.

Ceci implique :

- la minimisation du nombre et de la complexité des calculs (multiplications et divisions autre que des décalages prédiction des expressions réelles...etc)
- l'ordonnancement des calculs
- le calcul des facteurs par anticipation.

Dans le cas du codage Télévision, le "*Temps réel*" est très contraignant puisqu'à la fréquence d'échantillonnage (12 Mhz) (respec. 13.5 Mhz "*dernière norme numérique*") correspond un temps de cycle de 83ns (resp. 74ns) pendant lequel le bloc "*prédiction*" du système de codage doit produire le prédicteur du point courant.

2-3 : - Contrainte "Hardware"

En plus des problèmes de choix de technologie imposée par la contrainte de Temps réel, reste le souci de minimiser le volume d'informations à mémoriser en ce qui concerne le passé de l'image ou de la séquence. Dans le domaine interimage, une mémoire d'image ou au minimum de trame est indispensable.

La modélisation par vecteur d'état (décrite en partie II) permet de minimiser au mieux le volume-mémoire en résumant l'ensemble du passé à un vecteur de paramètres d'état

III - APPROCHE ADOPTÉE

3-a : - But et contexte de la thèse

Le but de cette étude est la construction d'un estimateur de mouvement adapté au codage avec compensation de mouvement par une séquence d'images de télévision.

L'estimateur de mouvement construit doit être capable de traiter n'importe quel type de scènes :

- plusieurs objets présents dans la scène
- aucune information à priori sur eux
- recouvrement possible des objets les uns sur les autres

et n'importe quel type de mouvements :

- présence de plusieurs mouvements d'amplitudes bien distinctes
- "mouvement" de translation, rotation mais aussi de déformation (cf Zoom)

Le manque d'information à priori sur le contenu de la scène interdit toute segmentation préalable avant l'estimation du mouvement proprement dit.

Les deux remarques précédentes privilègient pour le choix de la classe d'estimateurs visés, les estimateurs adaptatifs comme ceux décrits en 1-2b.

$$\vec{D}_{\text{point courant}} = \vec{D}_{\text{point précédent}} + (\text{gain} \times f_{\text{correction}})$$

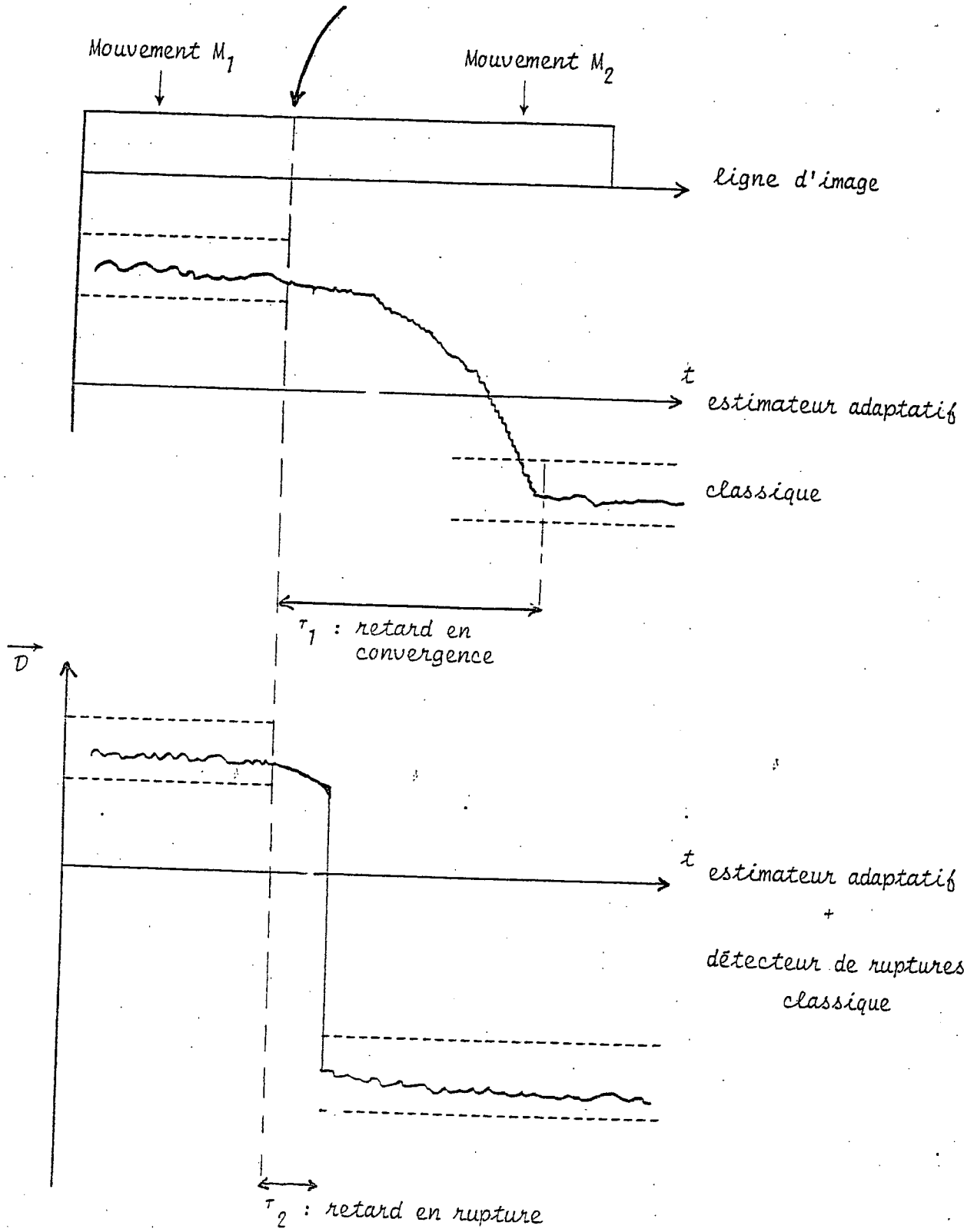
avec $f_{\text{correction}}$ dépendant causalement du passé spatial et temporel.

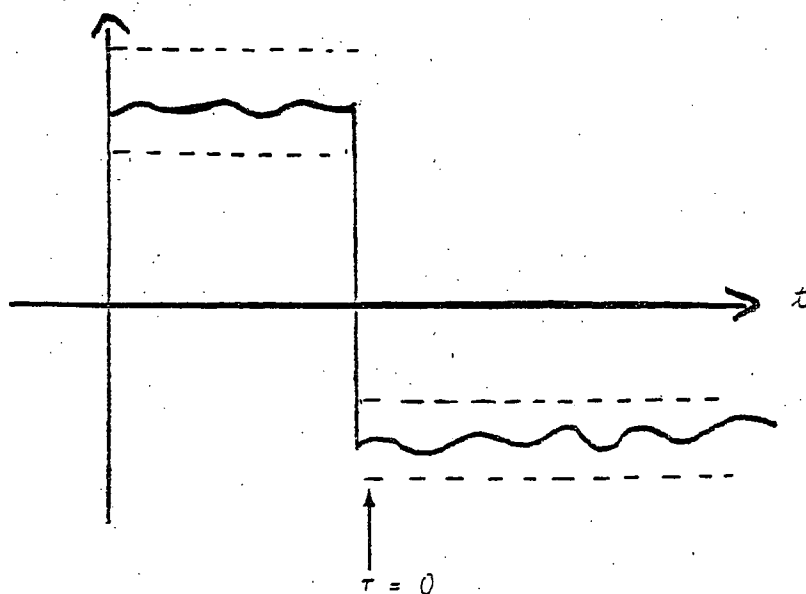
Cependant les estimateurs adaptatifs classiques réagissent avec un temps de retard (cf schéma 3.a) en présence de nouvelles observations très différentes des précédentes. Ceci peut se produire en image lorsque dans la scène existent plusieurs objets à mouvements forts distincts ou tout simplement entre le fond fixe et un objet en mouvement d'amplitude élevé. Ce retard peut être minimisé par un choix optimal du gain mais il reste le compromis à résoudre entre un gain élevé assurant un retard en convergence plus faible et un gain faible assurant un bruit d'agitation résiduelle minimum en régime stationnaire.

Il convient donc d'ajouter à l'estimateur adaptatif un détecteur de changements brusques de mouvement afin de forcer la réinitialisation de l'estimation à une valeur fonction de ces nouvelles observations.

$$M_1 \neq M_2$$

Changement brusque de Mouvement





estimateur
adaptatif +
détecteur "instantané" de ruptures.

pas de retard à la détection

Schéma 3.a

Les détecteurs de rupture classiques possèdent un retard à la détection puisqu'en règle générale, il faut le temps de constater un désaccord du mouvement courant avec le modèle précédemment ajusté. Pour pallier cet inconvénient, il faut en plus du schéma adaptatif d'estimation mener en parallèle un détecteur de ruptures basé sur des observations et des tests indépendants du processus d'estimation lui-même et dont on dispose immédiatement.

3-b : - Information a priori sur les ruptures de mouvement

L'hypothèse choisie consiste à dire que les changements brusques de mouvement dans une image sont situés sur les frontières des objets.

Il vient donc que les zones-ruptures de l'estimateur (au sens du paragraphe précédent) sont localisées sur les contours.

Pour vérifier cette hypothèse, il suffit d'examiner une image-différence temporelle point à point ($\forall x, I(x, t) - I(x, t - \tau)$): l'ensemble des contours d'objets en mouvement est inclus dans l'ensemble des points à différence temporelle importante.

Cependant les ruptures de l'estimateur ne doivent être faites que sur les "véritables" contours d'objets en mouvement (par exemple les contours suffisamment longs et minces) ; nous voulons différencier de cette manière les points-contours de texture et les frontières véritables d'objets. Ceci amène à la définition de règles de validation sur les contours afin de les classer comme contours d'objets en mouvement et donc zones ruptures pour l'estimateur ou non. Le schéma "3b" résume les différents modules à définir :

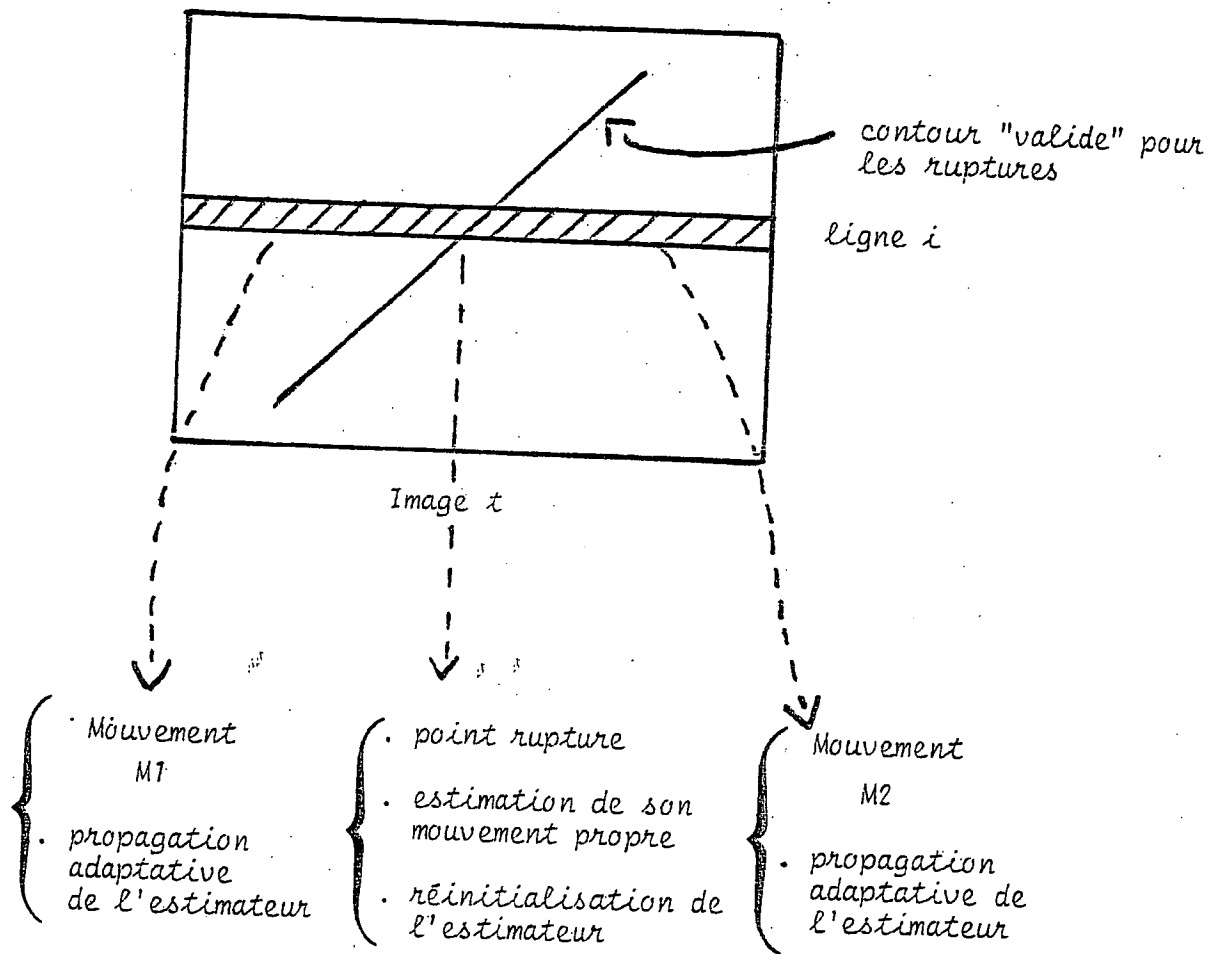


Schéma 3.b

L'approche adoptée demande donc la définition de trois modules (outre le traitement spatial pur de la détection, suivi et paramétrisation d'état des contours spatiaux présents):

- *l'estimation du mouvement propre de contours*. Cette information servira à la réinitialisation de l'estimateur si le contour concerné est déclaré zone-rupture

- *définition de règles de validation spatiale de contours* afin de classifier ceux-ci en zone-rupture ou non

- *Etude de la propagation de l'estimation* : quelle fonction de correction choisir dans l'algorithme adaptatif d'estimation ?

3-c : - Présentation des deux chapitres suivants :

Après un rappel et une extension de l'algorithme de détection et suivi spatial de contours utilisé (algorithme *Richard* [33,34]), nous décrirons deux algorithmes d'estimation du mouvement de contours :

- * - (MC1) : une méthode complète mais lourde et coûteuse en mémorisation qui consiste à extraire d'un algorithme de suivi de contours spatio-temporels l'information-mouvement qui nous intéresse dans le cadre de cette étude.

- * - (MC2) : le deuxième algorithme au vu des résultats de (MC1) inclut l'estimation du mouvement des contours dans leur suivi spatial. Ceci consiste en résumé à augmenter le vecteur d'état qui modélise le contour par des paramètres "*temporels*" capables de nous fournir une information mouvement.

Le troisième chapitre compare divers algorithmes adaptatifs d'estimation avec ou sans rupture sur les contours d'objets. L'algorithme final d'estimation inclut les trois modules cités en 3-b avec des versions simplifiées dans le souci d'une implantation future "*Hardware*" la moins coûteuse possible.

2eme PARTIE

DETERMINATION DU MOUVEMENT DES CONTOURS

II - (1) : TRAITEMENT DES DONNEES

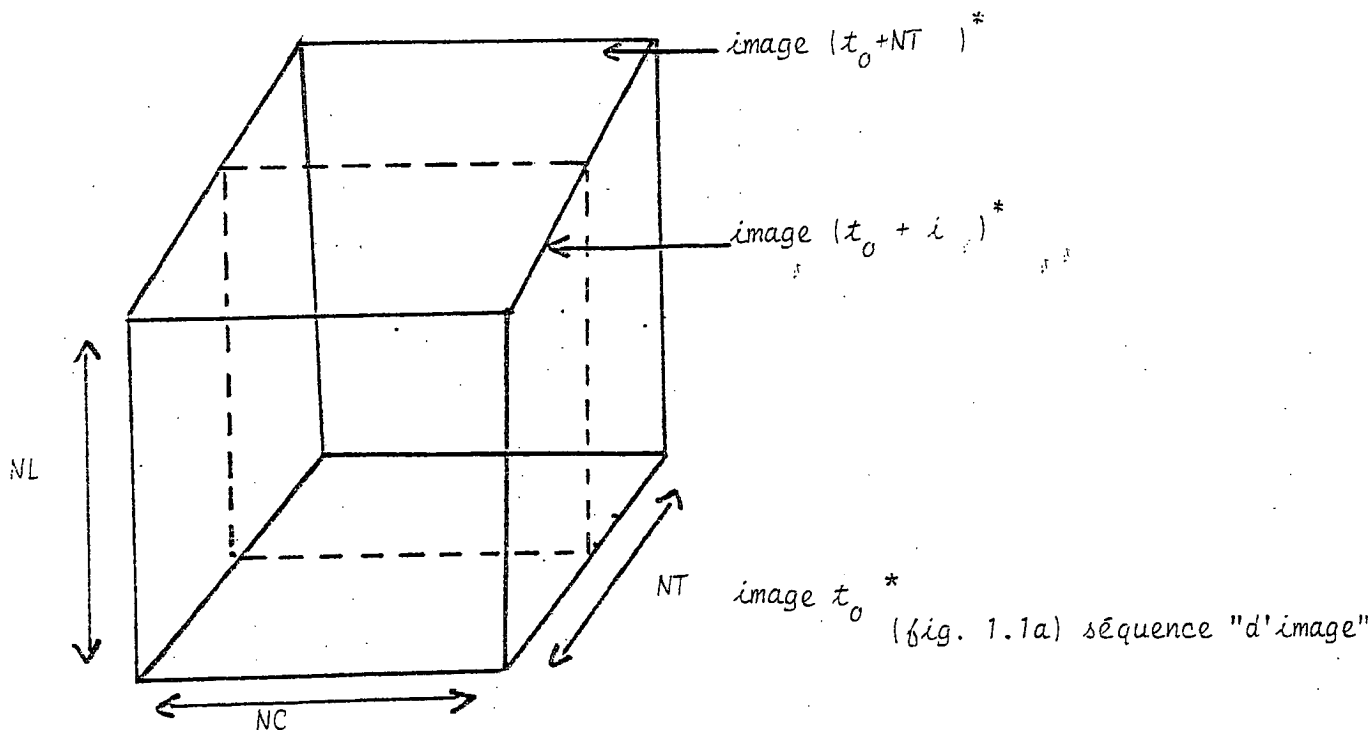
II (1.1) : Séquence d'images

Une séquence d'image peut être matérialisée sous forme d'un "cube" d'intensités de format $(NL \times NC \times NT)$

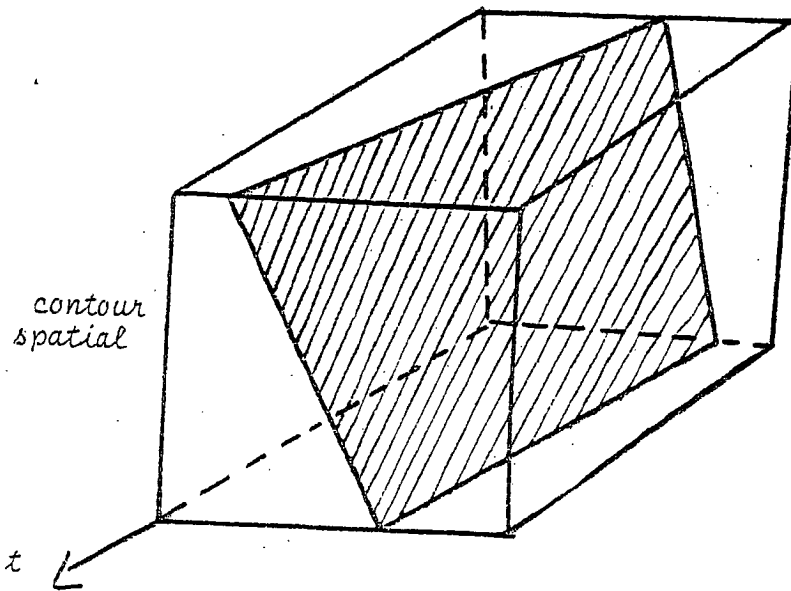
$\left\{ \begin{array}{l} NL : \text{nombre de lignes d'une image} \\ NC : \text{nombre de colonnes d'une image} \\ NT : \text{nombre d'images de la séquence} \end{array} \right.$

L'ensemble des données dont on disposera séquentiellement dans le temps est donc un espace de dimension 3 formé de l'ensemble des luminances $I(l, k, t)$

$\begin{array}{ccc} \uparrow & \uparrow & \uparrow \\ n^{\circ} \text{ligne} & n^{\circ} \text{colonne} & n^{\circ} \text{d'image} \end{array}$



* : désigne la période d'échantillonnage temporel choisie : * désigne la période-image (40ms) ou période-trame (20ms) suivant que le traitement interimage envisagé est un traitement trame à trame et/ou image à image.



"contour" spatio-temporel

L'introduction d'une seconde dimension pour la définition des contours provoque une complexité de suivi et de mémorisation beaucoup plus grande. De plus l'ajout d'une telle dimension ne corrige en rien la non-isotropie de l'estimation des mouvements.

Ce mode (4) sera utilisé par la suite dans l'algorithme MC1bis d'estimation du mouvement des contours.

II (1.2) : Définition et détection d'éléments de contours (EC) en mode spatial

Soit t_c , l'image étudiée. Avec les notations pour les intensités $I(k, l, t_c) = I(k, l)$, nous définissons la présence d'éléments de contours entre deux points d'image comme suit :

élément de contour horizontal (EC_h) si

$$|I(k,l) - I(k,l-1)| \geq \lambda_H$$

élément de contour vertical (EC_v) si

$$|I(k,l) - I(k-1,l)| \geq \lambda_v$$

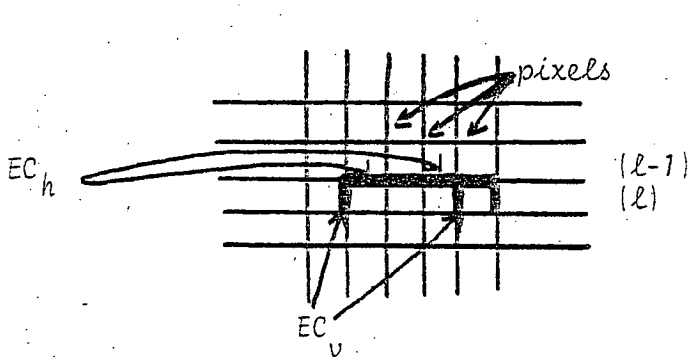


Fig. 1.2a : Connexité

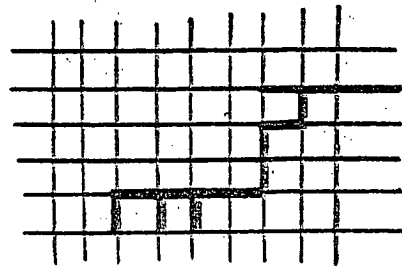


Fig. 1.2b : contour

Une connexité présente à la ligne l d'image sera donc constituée d' EC_h (entre la ligne courante l et la ligne précédente $(l-1)$) et d' EC_v (entre des points de la ligne l) connectés entre eux.

Les seuils de gradients verticaux (λ_v) et horizontaux (λ_h) dépendent du contraste présent dans l'image. A titre d'exemple ils ont été choisis ainsi : pour la séquence "Judith" : $\lambda_h = \lambda_v = 15$

pour la séquence "Chateau" : $\lambda_h = \lambda_v = 25$

(séquence d'images numérisées 8 bits).

Ces seuils devront donc être considérés comme paramètres ajustables suivant le type d'image (ou de qualité d'image).

Un contour sera constitué d'un nombre minimum ($N_{min} = 2$ ou 3) de connexités connectés entre elles ligne à ligne.

En réalité, sur l'ensemble des algorithmes de suivi de contours présentés par la suite, le suivi se fera non pas connexités par connexités mais marches à marches, le terme *marche* pris au sens de la figure (1.2c) . Ceci permettra un pré-filtrage des mesures locales (tels le signe, l'orientation des contours) si la marche est constituée de plusieurs connexités

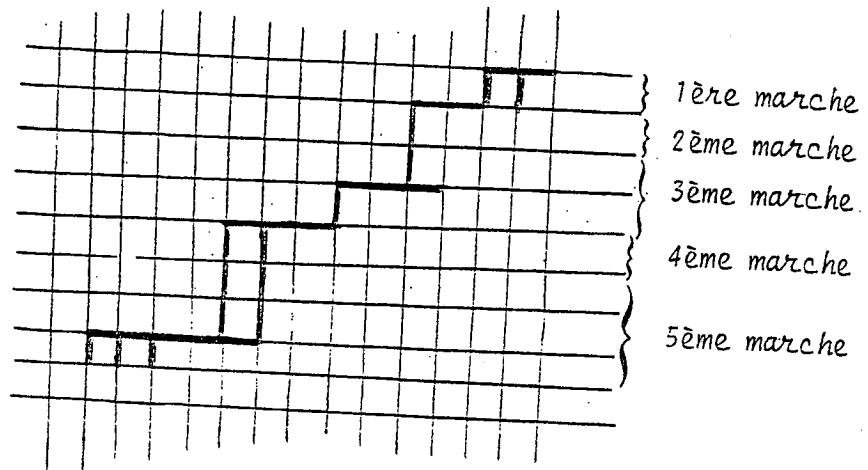


Fig. 1.2c : Contour décrit par marches

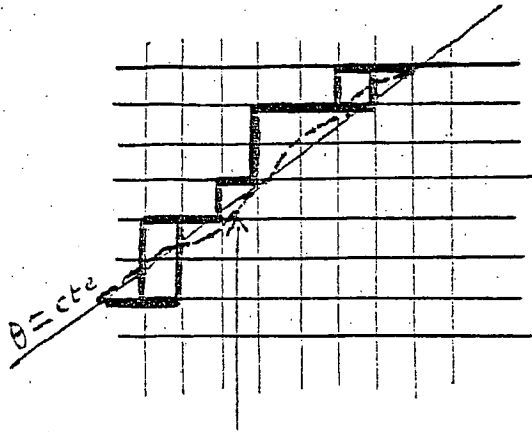
L'ensemble de ces définitions est détaillée dans [33] , [34] .

Modèle de contour associé :

Le modèle de contour choisi est un segment de droite dont l'orientation est constante à un bruit additif près (Figure 1.2d)

$$\underline{\theta = cte + bruit}$$

Ce modèle permet d'éliminer les gradients de "texture" détectés, ceux-ci ayant une orientation bruitée fortement



observation

Fig. 1.2d : contour

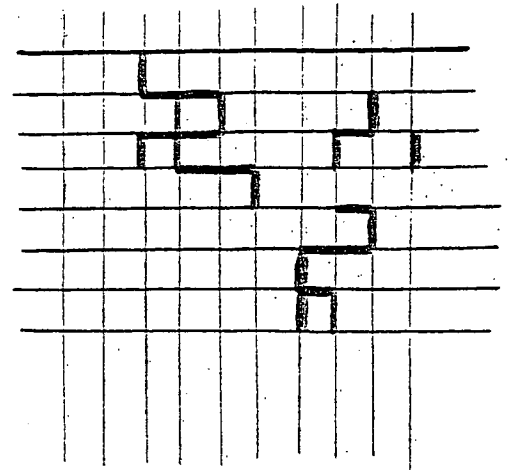


Fig. 1.2c : texture

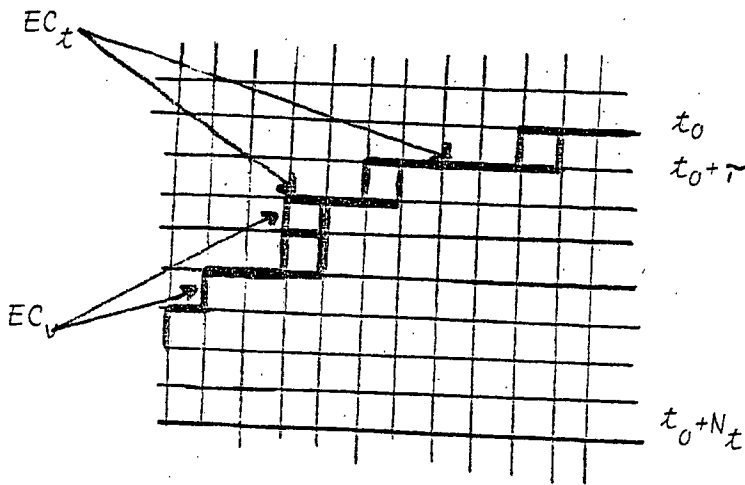
II (1.3) : Définition de contours spatio-temporels

La détection de contours par connexion d'éléments de contours horizontaux (EC_h) et verticaux (EC_v) développée dans le paragraphe précédent dans le cadre du mode (1) purement spatial, peut s'appliquer d'une manière identique aux deux autres modes (2) et (3) spatio-temporels.

Il faut noter toutefois que les gradients calculés dans ces deux modes de traitement des données (2) et (3) sont de deux natures bien distinctes (cf Figure 1.3a,b) :

- des gradients spatiaux (EC_h ou EC_v suivant respectivement le mode (3) ou (2) choisi)
- des gradients temporels (EC_t) : éléments de contour temporels)

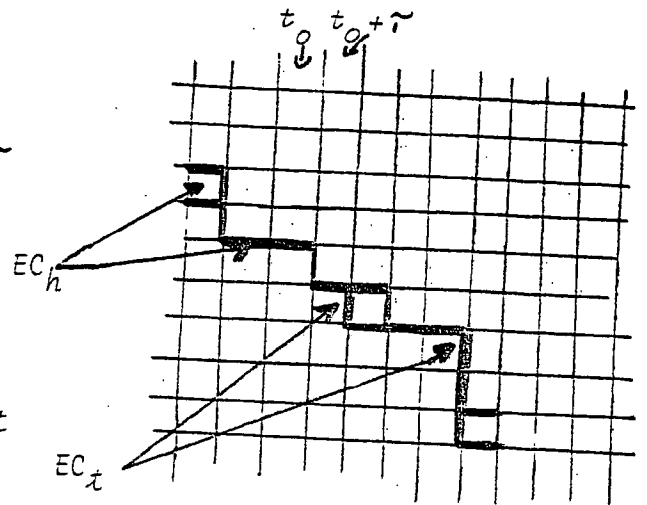
Les seuils de gradients spatiaux (λ_h, λ_v) et temporel (λ_t) peuvent être différents puisqu'ils correspondent respectivement à des différences intra et inter-images.



Mode (2)

Suivi temporel de la ligne l_c

fig. 1.3a



Mode (3)

suivi temporel de la colonne k_c

fig. 1.3b

Remarques

- Pour les raisons exposées précédemment (cf II (1.1)), nous délaierons les informations d' EC_t du mode (3) en gardant à l'esprit qu'ainsi nous perdrons une information à priori sur le mouvement vertical (en y : Δy) sur les éléments de contours horizontaux.

- l'information mouvement contenue dans le mode suivi-temporel-ligne est une information à priori sur le déplacement horizontal (en x : Δx) des éléments de contours verticaux, en prenant l'hypothèse que les EC_v reliés temporellement appartiennent au même contour dans deux images différentes (cf Figure 1.3c)

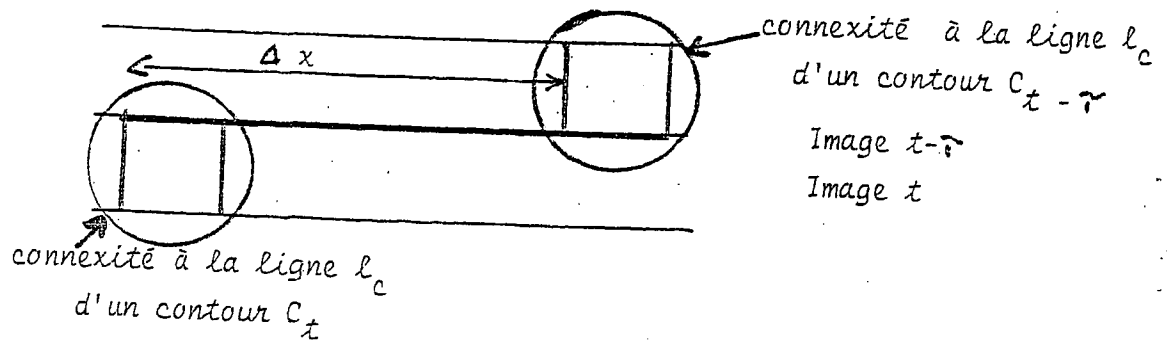


Fig. 1.3 c

Ce suivi temporel ligne ne tient pas compte du glissement propre du contour C sur lui-même (cf Figure 1.3d). Le vecteur de déplacement réel (D_x, D_y) du contour C n'est connu, par les méthodes de suivi des modes (2) et (3), que par ses projections Δ_x et Δ_y . Une relation simple existe entre la projection Δ_x et le vecteur déplacement à norme minimale $\vec{D}_\perp (D_{x\perp}, D_{y\perp})$ si l'orientation du contour C est connue. Cette relation est donnée par la formule suivante :

$$\vec{D}_\perp = \begin{pmatrix} D_{x\perp} \\ D_{y\perp} \end{pmatrix} = \begin{pmatrix} \cos^2 \theta \cdot \Delta_x \\ \sin \theta \cdot \cos \theta \cdot \Delta_x \end{pmatrix} \text{ avec } \theta : \text{orientation du contour } C$$

La perte d'information due à la seule observation de Δ_x est exprimée dans la relation liant les paramètres Δ_x , D_x , D_y et θ :

$$\Delta_x = D_x - D_y \tan \theta$$

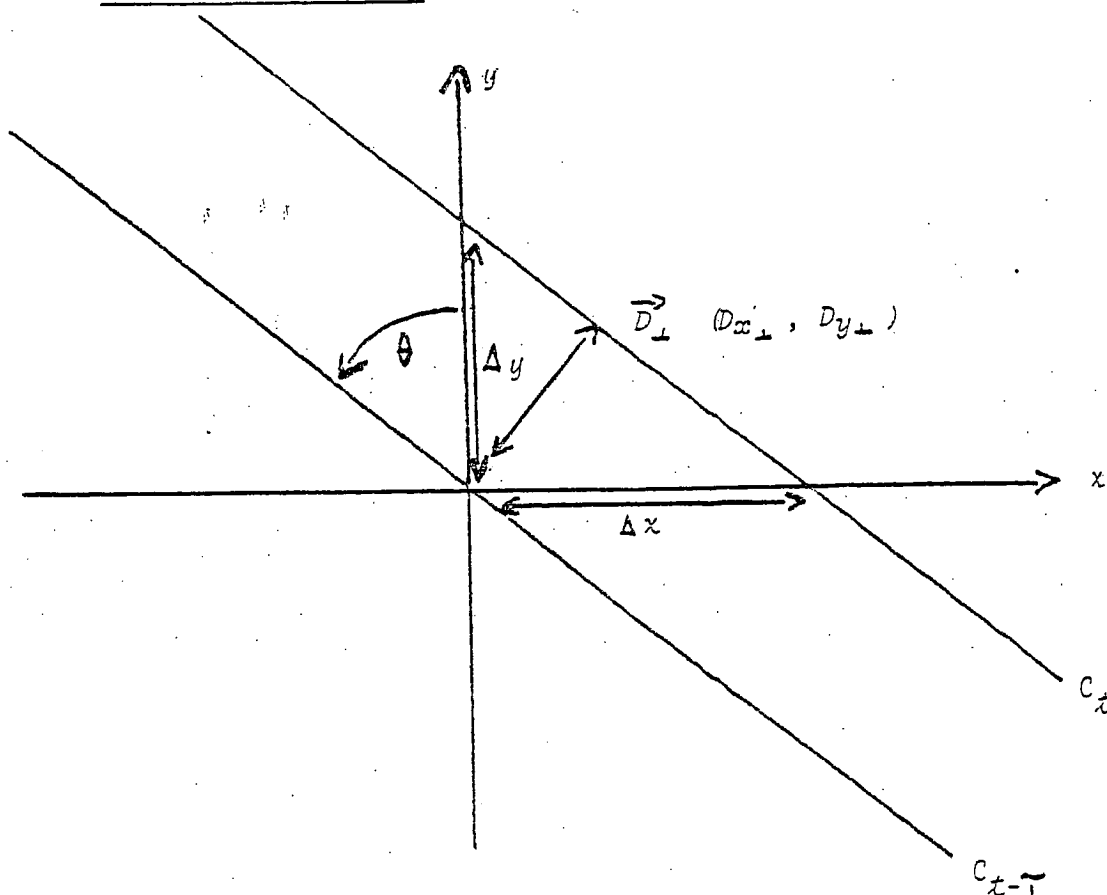


Fig. 1.3d

Notations :

- \vec{D} : déplacement réel du contour C
- \vec{D}_\perp : déplacement à norme minimale
- Δx : information-mouvement du mode (2)
- Δy : information-mouvement du mode (3)
- C_t : contour C à l'image t (orientation Θ)
- $C_{t-\tau}$: contour C à l'image $t-\tau$ (orientation Θ)

Cette redondance entre paramètres a été vérifiée pour des contours d'orientation proche de $\pi/4$ ($\tan \Theta \simeq 1$). Aucune étude exhaustive sur l'ensemble des Θ n'a été menée.

* la figure (1.3c) représente le cas "idéal" de suivi temporel de connexités où la connexité de C_t n'est pas "déformée" entre l'image $t-\tau$ et où elle n'est reliée qu'à un seul contour de l'image $(t-\tau)$

La figure (1.3e) montre le cas d'une déformation temporelle (épaisseur de la connexité différente).

La figure (1.3f) montre le cas d'une "fourche" temporelle où la connexité de C_t est reliée à plusieurs connexités de divers $C_{t-\tau}$

Cette situation peut se produire lors de superpositions d'objets en mouvement inverses.

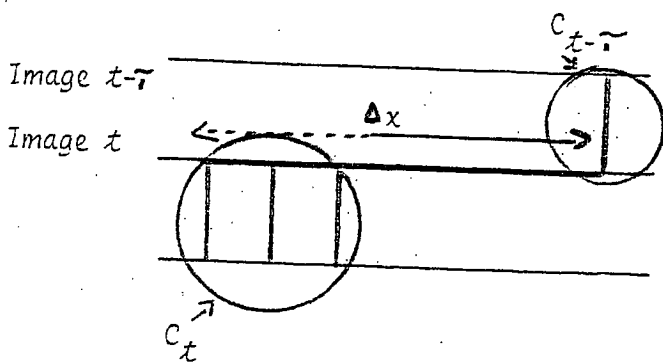


Fig. 1.3e

Déformation temporelle

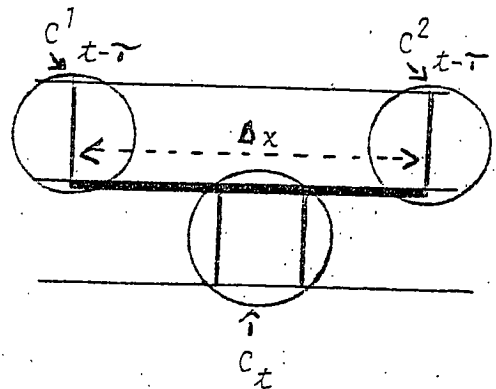


Fig. 1.3f

Fourche temporelle

Le calcul local de cette information-mouvement Δ_{∞} devra prendre en compte ces cas particuliers.

II - (2) : ALGORITHME DE SUIVI DE CONTOURS : *algorithme Richard* [33, 34, 30]

Cet algorithme de suivi a fait l'objet d'une précédente thèse [33] et a été implanté et testé sur maquette (*projet PREDITE* [35])

Nous rappelons ici les principales notations utilisées et adoptons pour l'algorithme lui-même une formulation par automates.

II (2.1) : présentation de l'algorithme

A chaque contour spatial de l'image est associé un automate d'état (à vecteur d'état fini) qui va fonctionner marche-à-marche le long du contour.

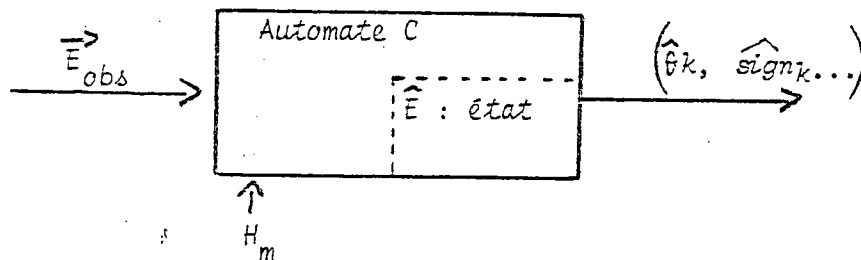


Fig. 2.1a : automate associé au contour C

i) description de l'automate :

* à la marche n°k du contour C, l'automate grâce au vecteur d'observations $\vec{E}_k \text{ obs}$ qu'il reçoit en entrée, réactualise l'état précédent \vec{E}_{k-1} en \vec{E}_k . La sortie est constitué par l'ensemble des paramètres utiles pour caractériser le contour : par exemple son orientation $\hat{\theta}_k$ et son signe $\hat{\text{Sign}}_k$

* *initialisation de l'automate* : à la première marche ($k=1$), l'initialisation de \vec{E} se fait de la manière suivante :

$$\vec{E}_1 = (1, H_1, (V_1 + P_1), 0, 0)$$

* *sortie de l'automate* : grâce aux composantes du vecteur d'état, divers paramètres caractéristiques du contour peuvent être estimés (en vue d'une prédiction adaptative, fonction des caractéristiques du contour).

Estimation de l'orientation

$$*) |\Theta|_k = f(\hat{H}_k, \hat{V}_k)$$

la fonction d'estimation f peut être : $|\Theta|_k = \hat{H}_k / \hat{V}_k$

ce qui donne une estimation de la tangente de l'angle (en valeur absolue)

Estimation du signe

$$*) \hat{\text{sign}}_k = f(\hat{F}_k)$$

la fonction d'estimation f est choisie égale à la fonction "sgn" définie par :

$$\text{sgn}(z) = \begin{cases} 0 & \text{si } z = 0 \\ |z| & \text{sinon} \end{cases}$$

ii) conflits : cas des fourches spatiales

Le problème se pose lorsque la marche courante k d'un contour est reliée de façon connexe à N contours d'origines différentes (c.a.d. auxquels sont associés N automates distincts). (cf figure 2.1c)

* si $N > 2$: on ignore les N automates des contours antérieurs et on considère la marche courante comme une marche initiale d'un nouveau contour ; un nouvel automate est donc initialisé (avec la procédure d'initialisation classique des automates décrite précédemment).

* si $N = 2$: les deux états internes des automates vont être "condensés" en un seul état d'un automate "équivalent".

L'état équivalent \tilde{E} sera construit par interpolation linéaire (cf annexe 2) des deux états \tilde{E}_k^1 , et \tilde{E}_k^2 .

La procédure de réactualisation de \tilde{E} par les nouvelles observations de $\vec{E}_{k_{obs}}$ se poursuit alors normalement : $\rightarrow \vec{E}_k$

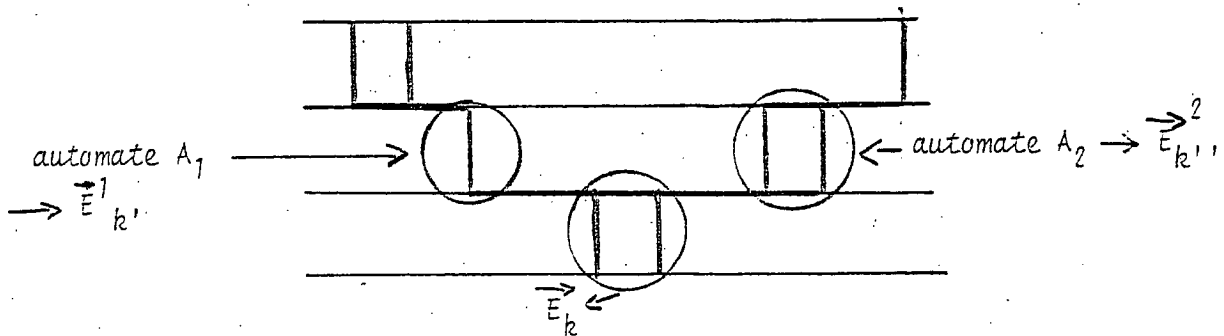


Fig. 2.1c "fourche" spatiale

iii) ruptures sur le modèle-contour

Deux types de rupture de modèle (cf II 1.2) sont possibles :

* rupture sur l'orientation

$$\text{si } \begin{cases} |\theta|_k = f(H_k, V_k) \\ \hat{|\theta|}_k = f(\hat{H}_k, \hat{V}_k) \end{cases} \quad \text{avec } f \text{ définissant l'estimateur d'orientation}$$

il y a rupture à la marche k du contour lorsque

$$| |\theta|_k - \hat{|\theta|}_{k-1} | > \lambda_{r-\theta}$$

dans les expérimentations et discrétisations choisies pour θ (cf annexe 1), $\lambda_{r-\theta}$ est choisi égal à 2

f)

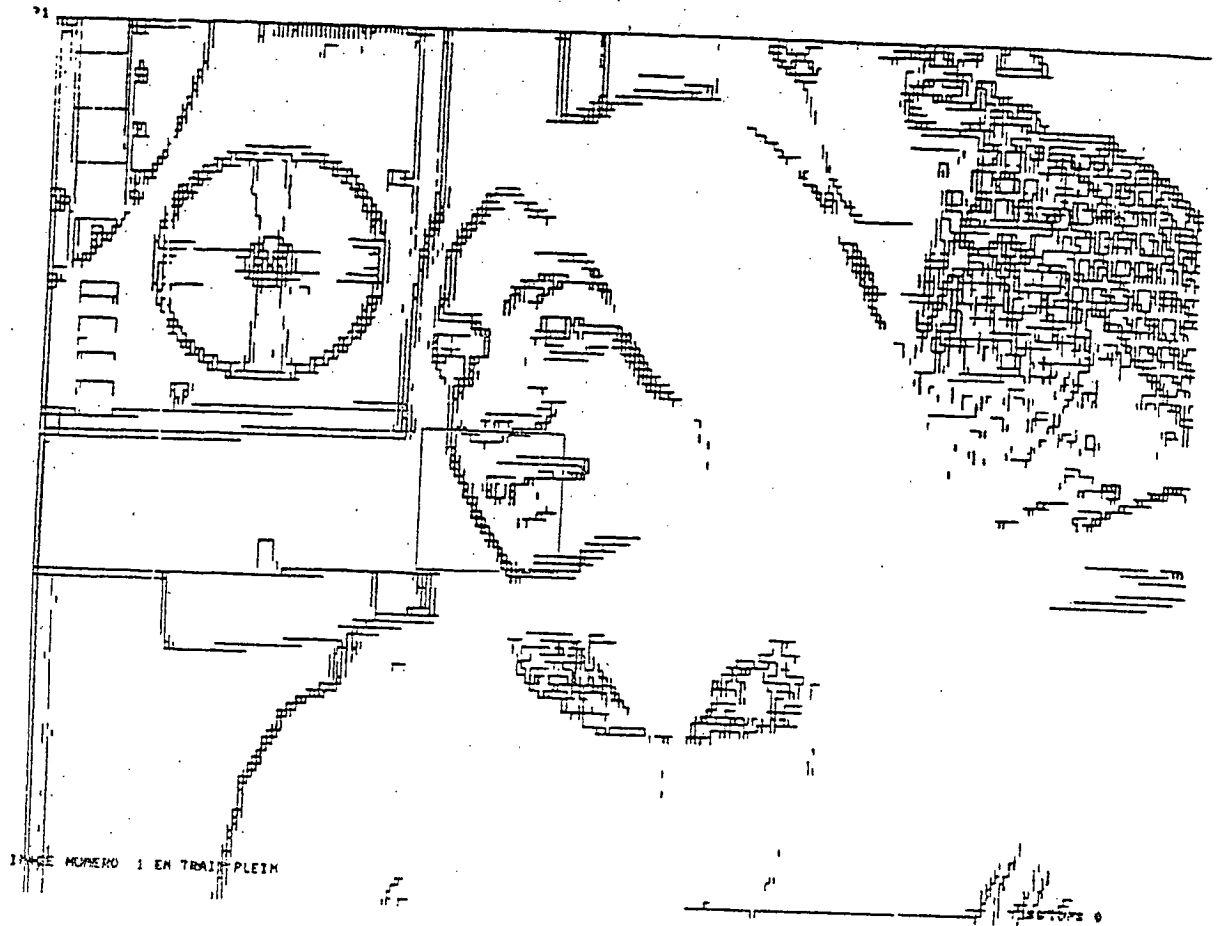


Fig. 2.2a - Image "Judith" $\lambda_v = \lambda_v = 10$

g)

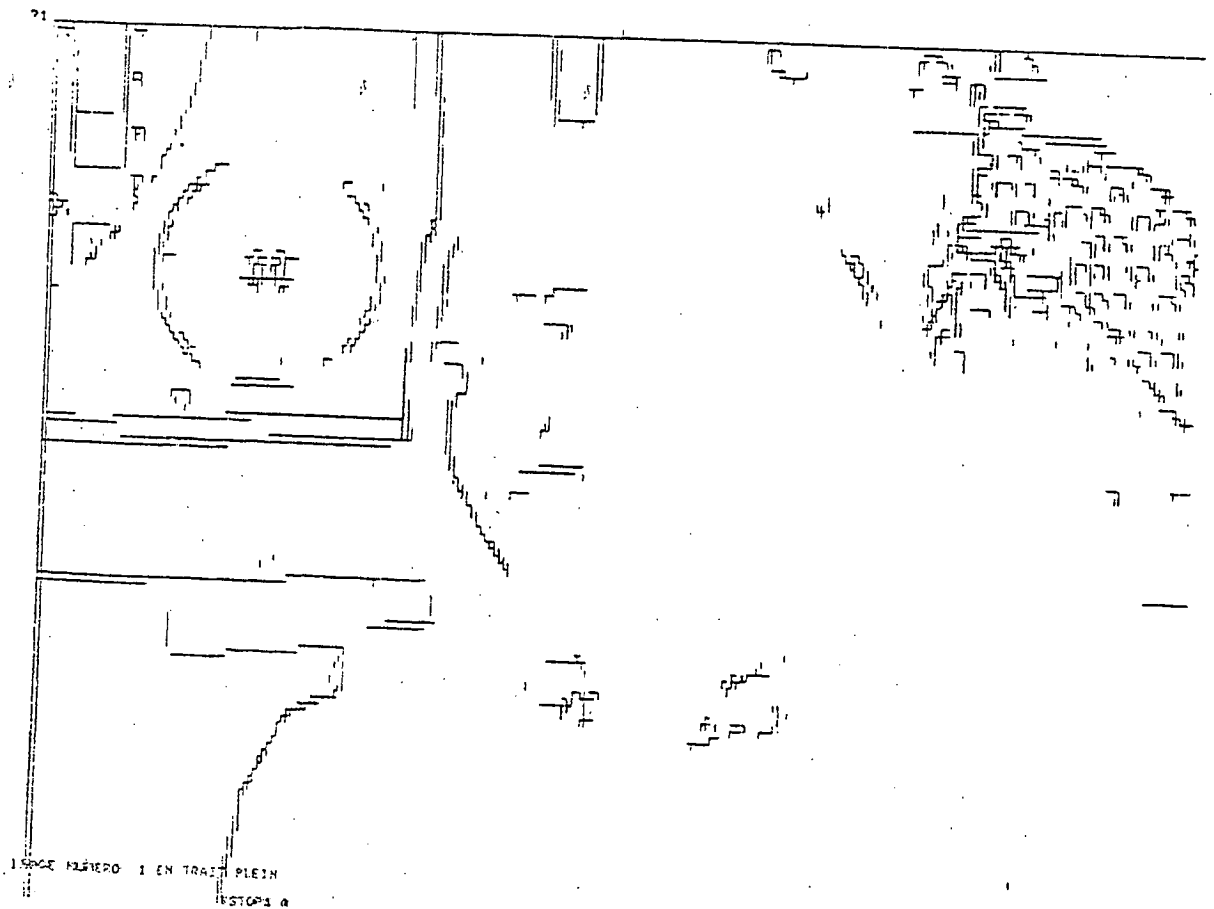


Fig. 2.2b - Image "Judith" $\lambda_h = \lambda_v = 20$

* rupture sur le signe : lorsque $\hat{Q}_k \geq 2$

Ce test n'autorise donc au maximum que 2 changements consécutifs de signe sur le contour.

Lorsque les tests de rupture sont validés, l'automate associé est réinitialisé.

II (2.2) : Résultats

. *Figure 2.2a* : Cette image représente l'image des gradients spatiaux d'une image de la séquence *Judith* (image n°31) avec pour seuils de gradient : $\lambda_h = \lambda_v = 10$

On y distingue plusieurs zones à caractéristiques différentes :

- la zone fixe avec à gauche la mire
à droite la zone "texture" du poster
- la zone "mobile" centrale constituée de :
 - . zones "textures" telles l'échancrure, la bouche, les yeux...
 - . et de zones "contours" telles la joue droite du visage, le front, les cheveux et l'épaule.

Les résultats suivants porteront sur la zone *JOUE* encadrée dans la figure.

. *Figure 2.2b* : même image que précédemment avec

$$\lambda_h = \lambda_v = 20 .$$

. *Figure 2.2c* : détail "macroscopique" de la zone "JOUE" extraite de la figure 2.2a

Cette zone *JOUE* comporte :

- une zone quasi-homogène à gauche du contour
- un contour bien net ($\theta = \pi/4$) constitué par le bord de la joue
- une zone "texture" (nez, bouche,...)

Nous allons examiner cette zone "JOUE" au travers de la séquence.

. Figure 2.2d : même figure que précédemment mais pour l'image suivante (n° image = 32 dans la séquence)

. Figure 2.2e : détection et suivi spatial de contours sur la zone "JOUE".

- l'itération (ITERC) portée sur le numéro d'image (ITERC = 1, image 31d)

- "100" désigne un point non contour

- "200" désigne un EC_h

Dans le cas d'un EC_v , l'orientation $\hat{\theta}$, sortie de l'automate associé, est affichée avec la quantification rappelée en annexe I

$$\left(\begin{array}{l} \hat{\theta} = 0 \text{ contour vertical} \\ \hat{\theta} = 8 \text{ contour quasi-horizontal} \end{array} \quad \hat{\theta} \simeq 4 \longleftrightarrow \pi/4 \right)$$

La corrélation temporelle d'une image à l'autre est particulièrement visible sur le contour de la joue (ce contour se déplaçant grossièrement d'un pel/image vers la gauche). En zone "texture" (cf bouche), les orientations spatiales trouvées dépendent de l'image.

Les paramètres de détection et du suivi sont les suivants :

$$\lambda_h = \lambda_v = 10 \quad ; \quad N_{\min} = 2 \quad ; \quad \lambda_{r\theta} = 2 \quad ; \quad K = 3$$

. Figure 2.2f : détection et suivi de contour sur une image de la séquence CHATEAU.

$$\lambda_H = \lambda_v = 25$$

Fig. 2.2e

Détection et suivi
de contours spatiaux

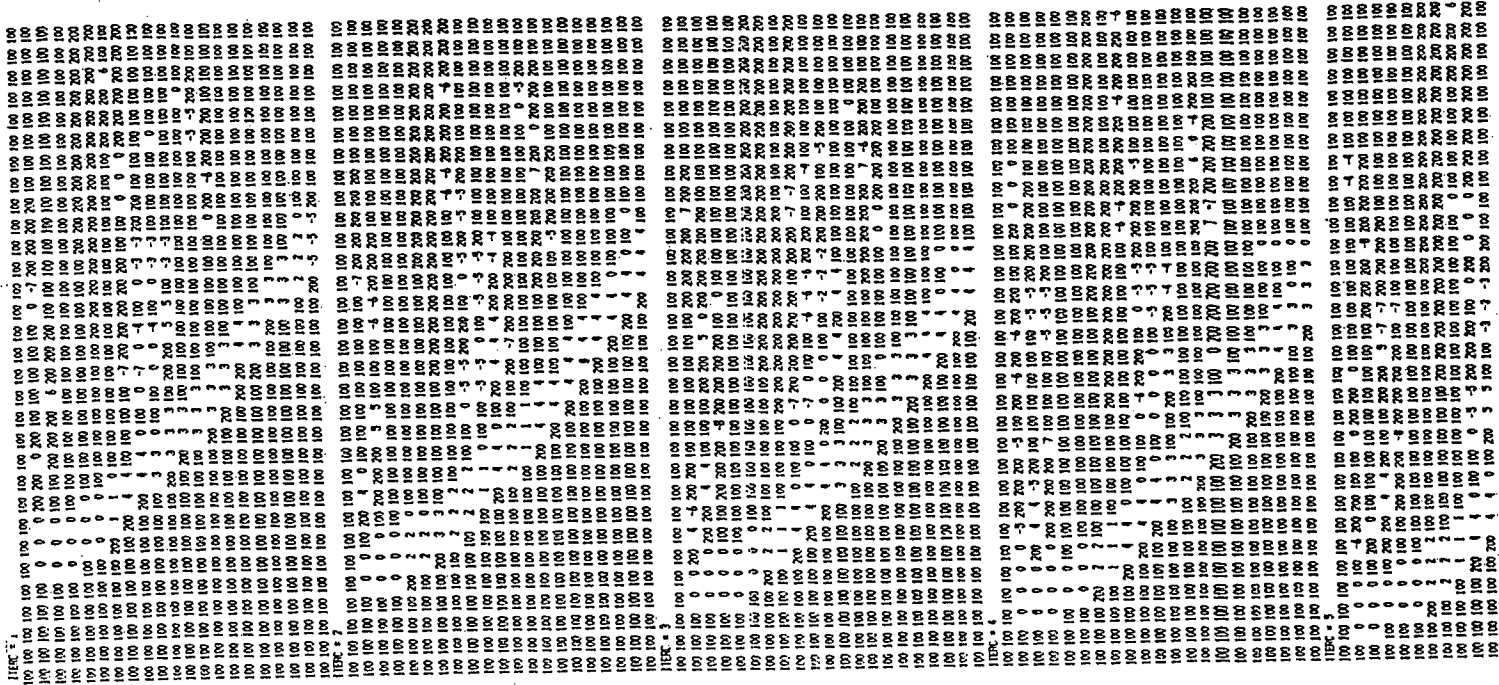
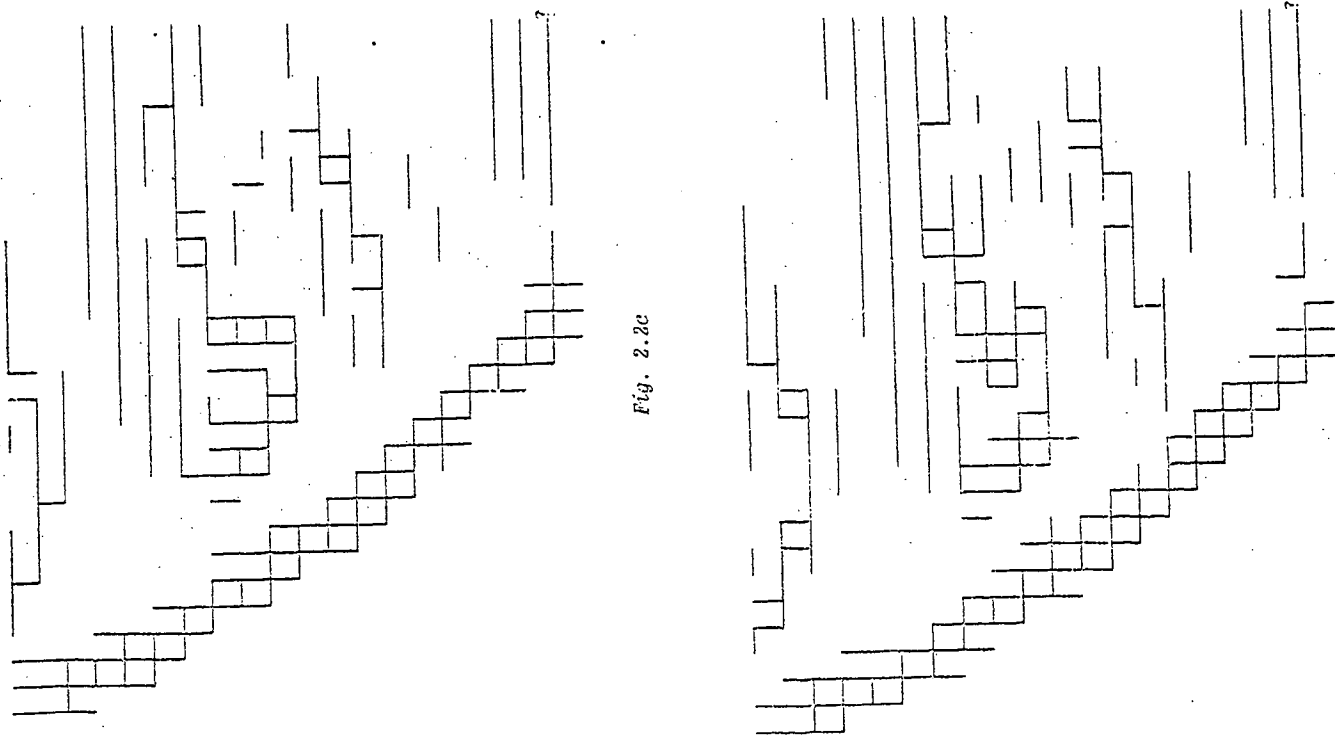


Fig. 2.2c



Ce paramètre d'ancienneté \hat{A}_k va permettre de mesurer l'"age" du contour indépendamment de la présence ou non de ruptures de modèles sur ce contour. Il estime en réalité la longueur de la ligne connexe matérialisant le contour.

Processus de validation de contours :

Rappel : Il s'agit ici de définir des règles de validation pour les contours qui seront considérés par notre estimateur du chapitre III comme zones ruptures pour l'estimateur (CF schéma I3.b)

La batterie de tests que nous définissons, visent à invalider les contours trop "épais" ou trop "court" (en vertical comme en horizontal)

Ces tests portent sur les paramètres d'état du contour.

pour les contours horizontaux

invalidation ssi

$$\boxed{\hat{H}_k < S_h}$$

S_h longueur minimale des contours horizontaux.

pour les contours non-horizontaux :

Invalidation si
et seulement si :
 $(\hat{V}_k > S_e)$ ou

S_e épaisseur maximale des contours

$$\left\{ \begin{array}{l} \max \left(\hat{H}_k, \left(\hat{V}_k + \hat{P}_k \right) \right) < S_l \\ \text{et} \\ \hat{A}_k < S_a \end{array} \right\}$$

S_l : "largeur" minimale des contours

S_a : ancienneté minimale des contours

Ceci permet d'éliminer à la fois les contours trop épais et les contours trop jeunes qui ne sont pas suffisamment larges.

Résultats :

* tab. 2.40 : a) détection de tous les gradients de la zone JOUE pour l'image Judith n°31 ; $\lambda_h = \lambda_v = 10$

b) $S_h = S_l = 10$; $S_e = 4$; $S_a = 5$

le seuil de rupture en θ est choisi à $\lambda_{r\theta} = 1$

seuls les gradients "longs" et réguliers sont validés (contours en trait plein)

c) même paramètres que précédemment sauf $S_a = 4$
la contrainte "ancienneté" est donc moins sévère

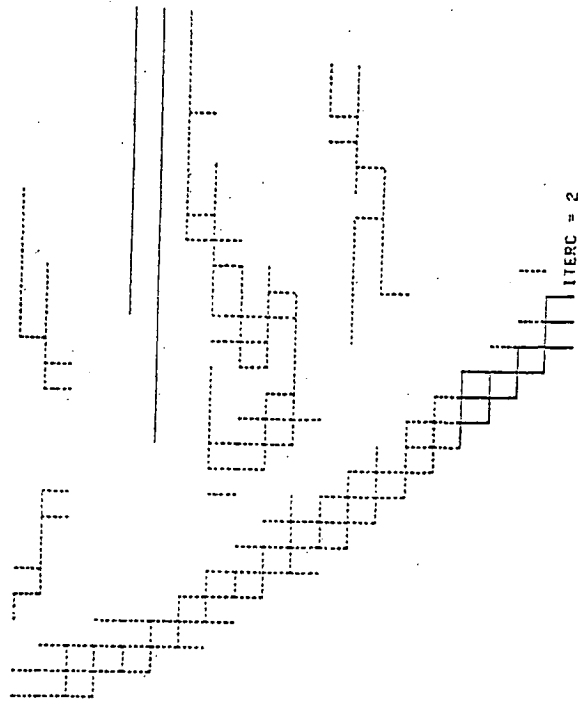
* tab. 2.41 : a) ; b) ; c) : même schémas et seuils que précédemment appliqués à l'image suivante (n°32)

Remarques

* les règles de validation peuvent être rendues plus sévères de deux manières différentes :

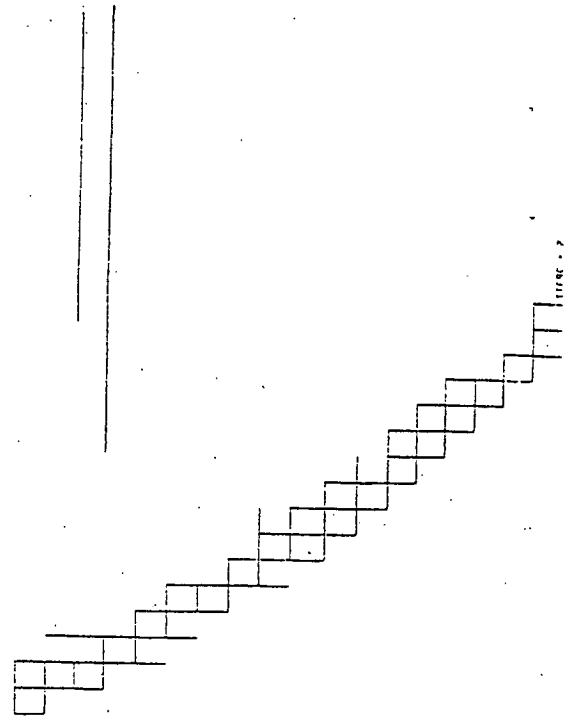
- d'une manière directe par augmentation des seuils S_e ; S_l ; S_h et S_a d'épaisseur, de largeur et de longueur minimale.
- d'une manière indirecte par diminution du seuil $\lambda_{r\theta}$ de rupture de modèle en orientation ce qui provoque des ruptures plus fréquentes et donc ralentit l'incrémentatation du paramètre d'état \hat{A}_k d'ancienneté. Les ruptures trop fréquentes ne permettent pas cependant une bonne convergence de l'estimateur d'orientation (θ) le long du contour.

* la règle de validation par l'ancienneté telle qu'elle a été décrite précédemment, nous fait perdre les premières marches du contour (cf Tab. 2.40b et 2.41b). Cet inconvénient peut être éliminé par une procédure de validation à *posteriori*

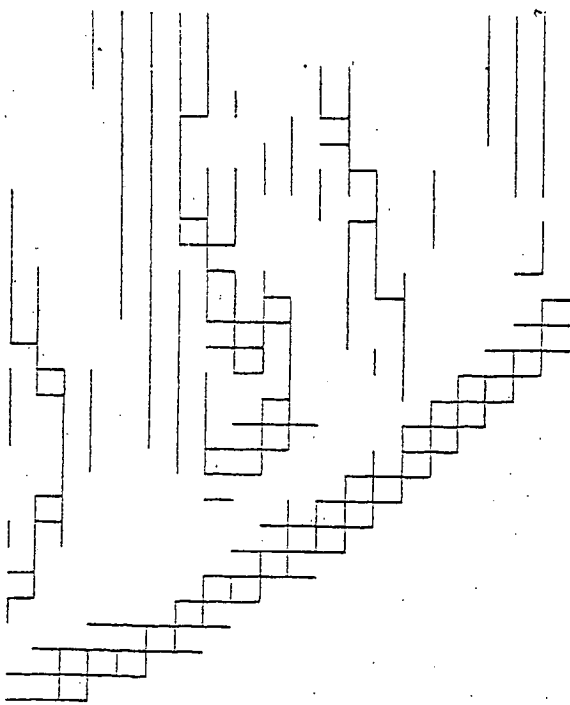


b)

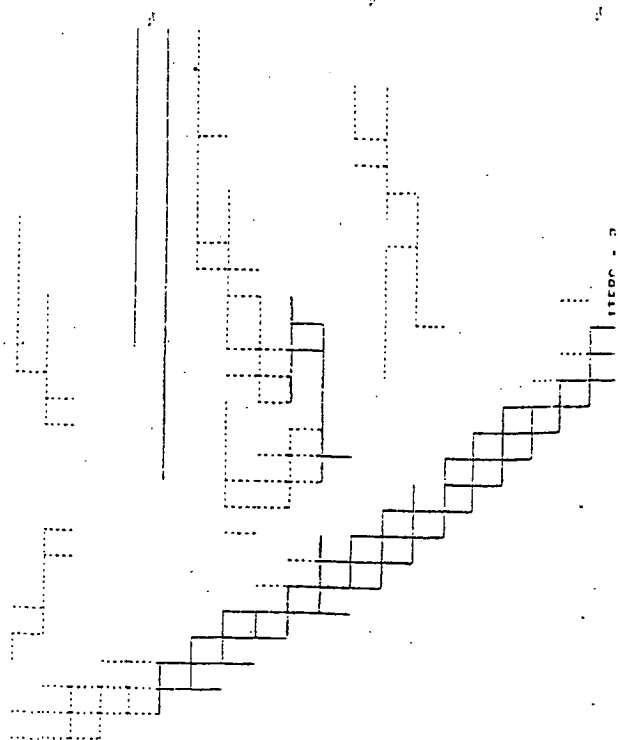
Tab. 2.40



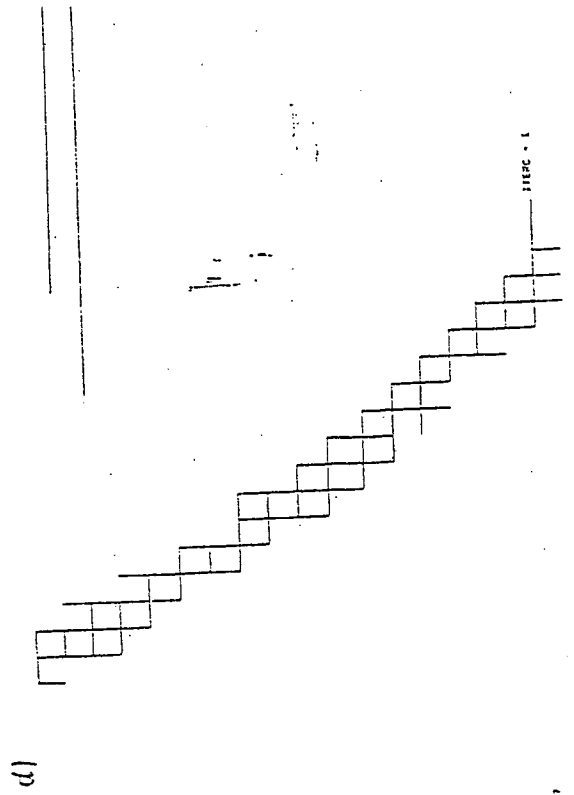
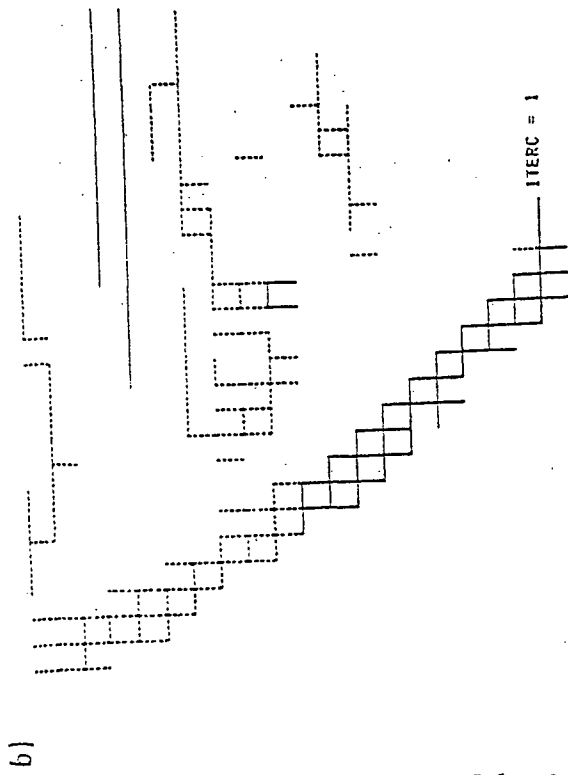
d)



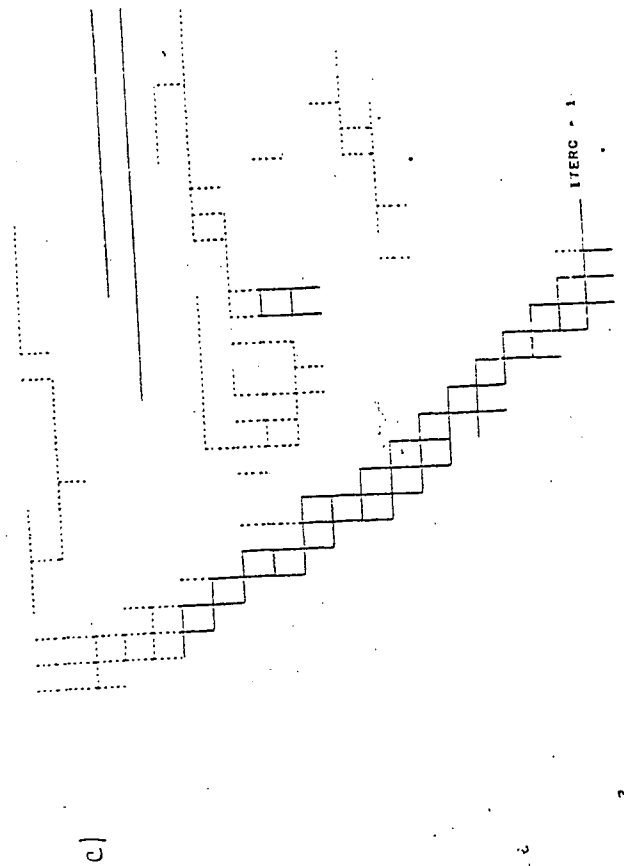
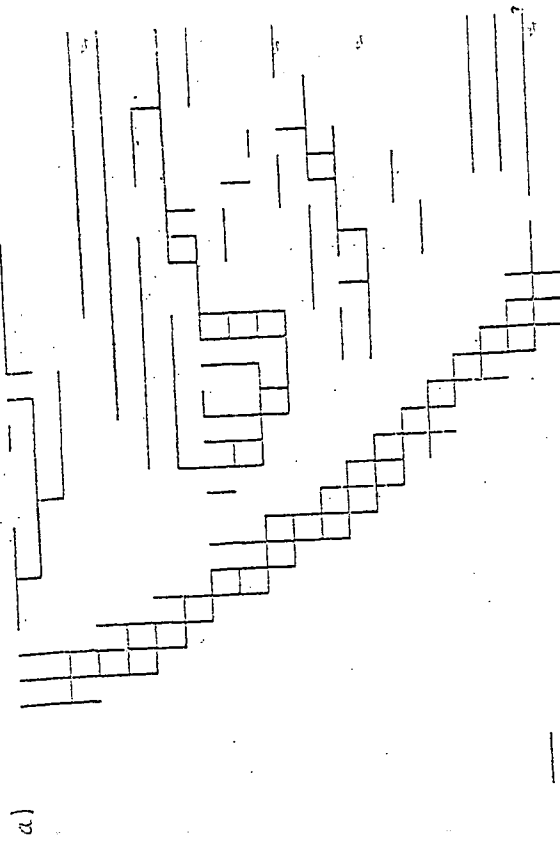
a)



c)



Tab. 2.41



a)

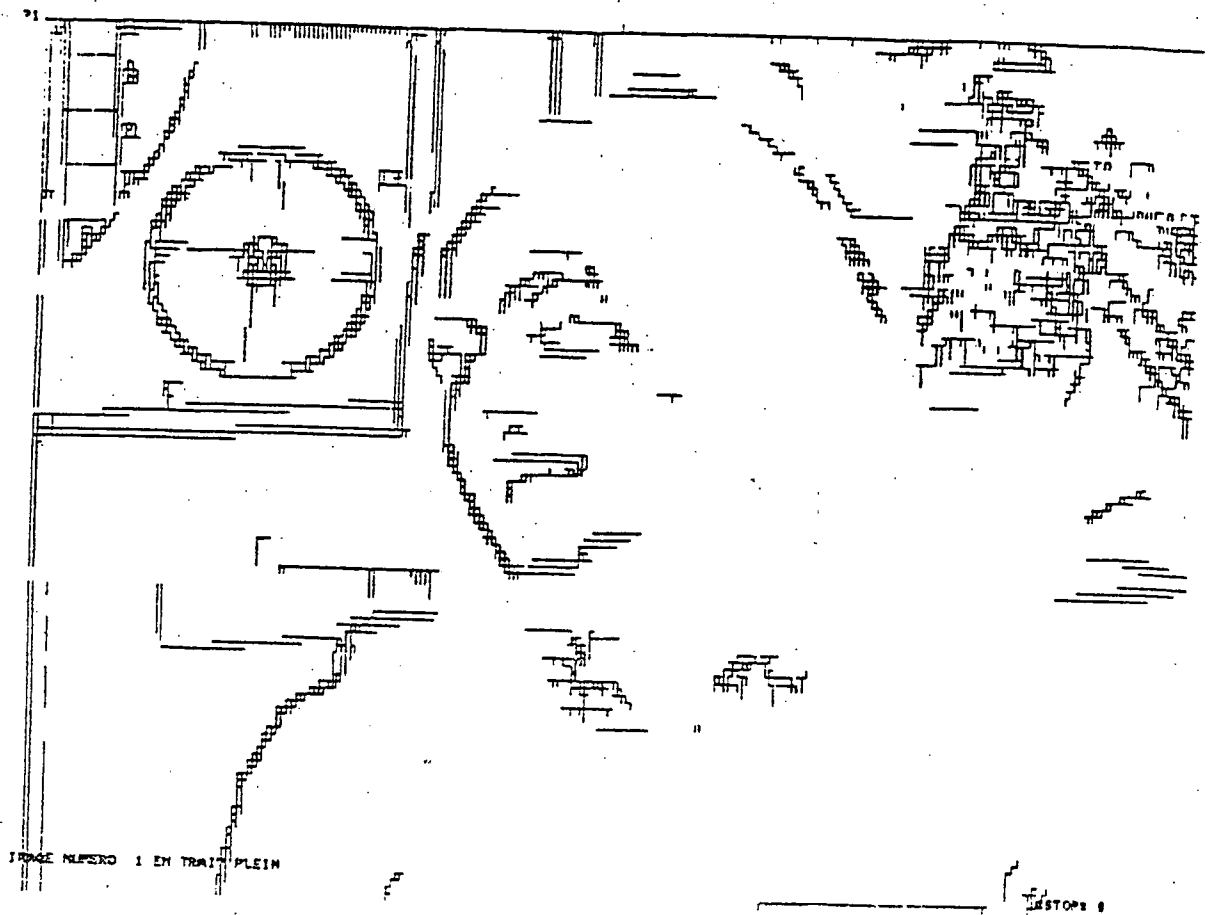
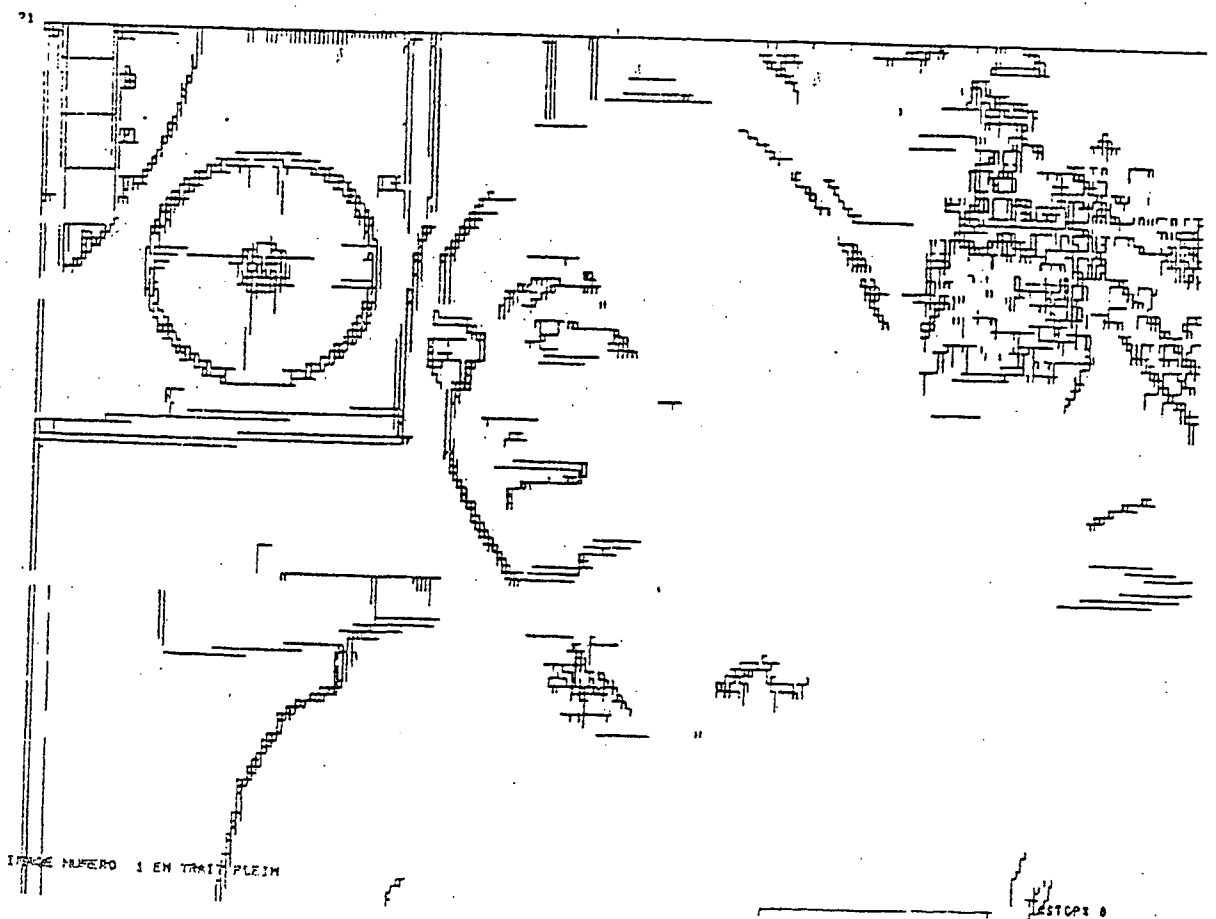
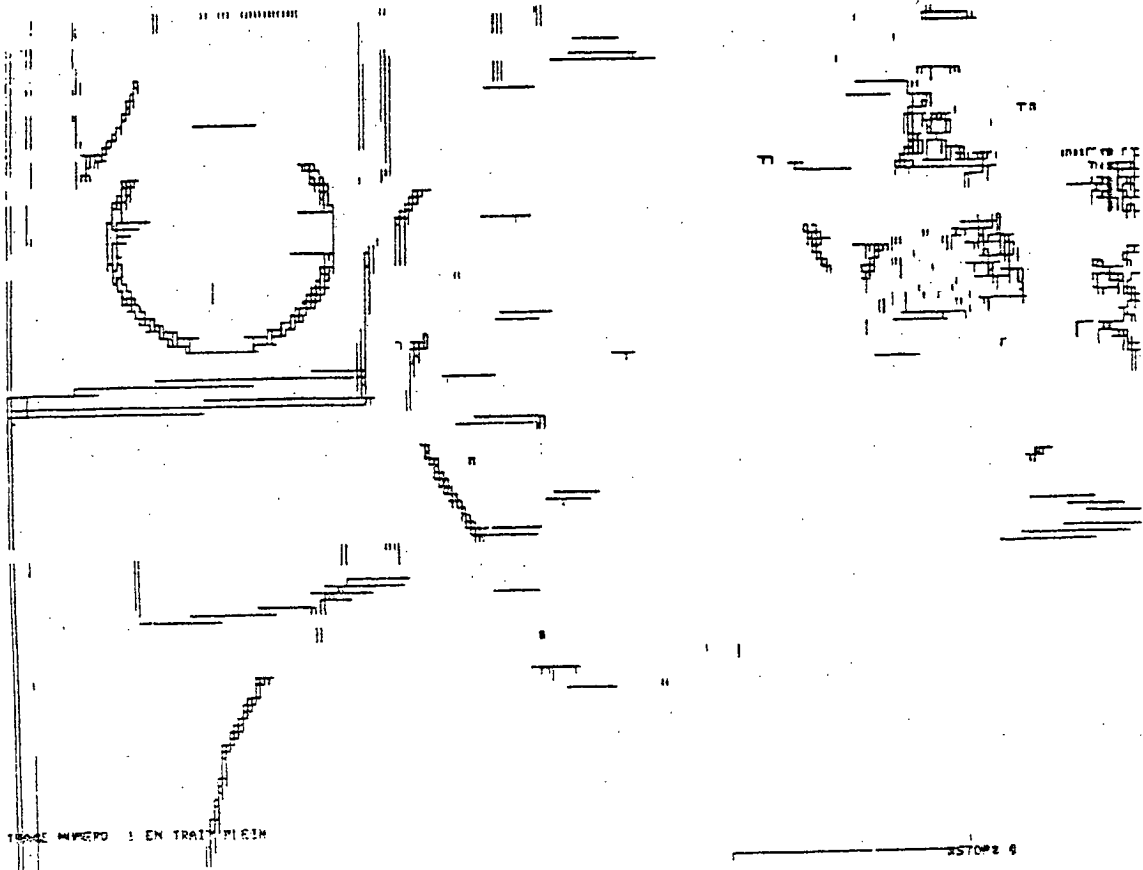


Tableau 2.6

b)

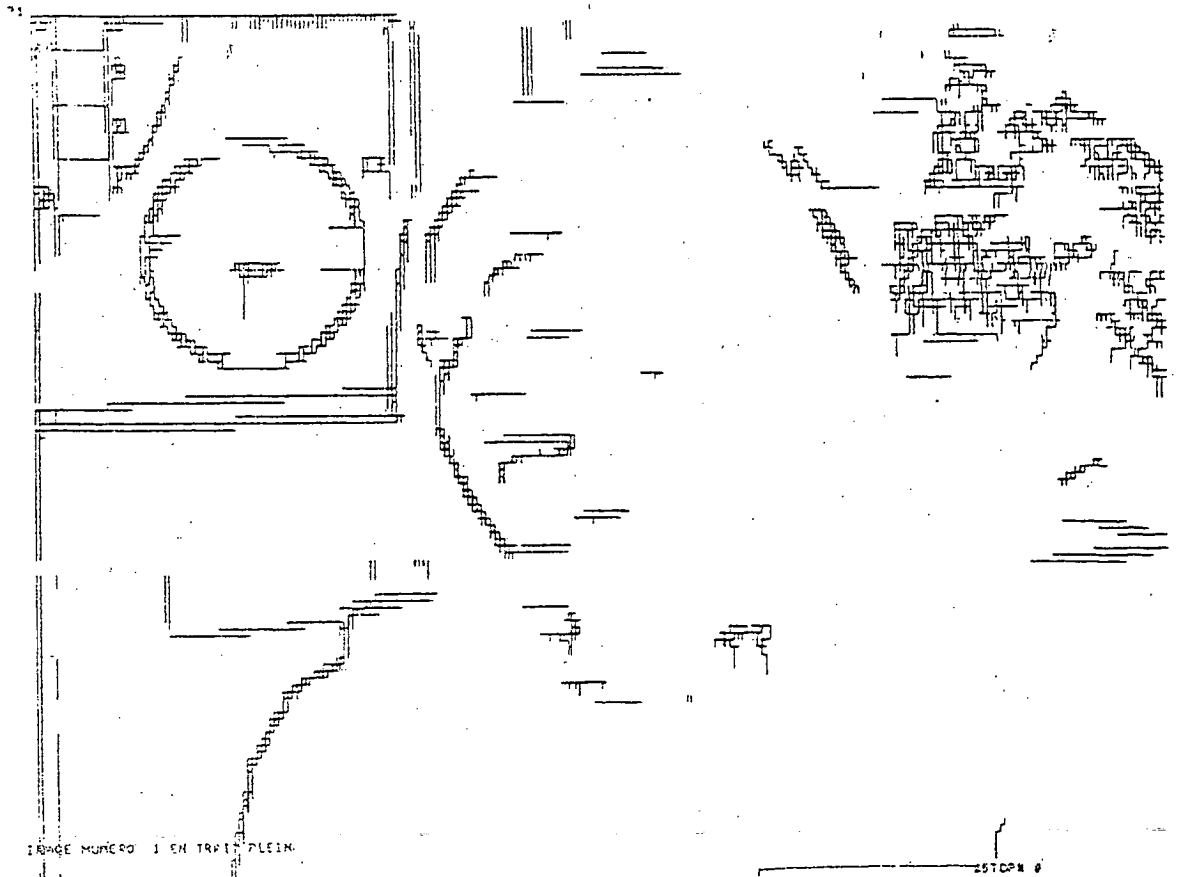


c)

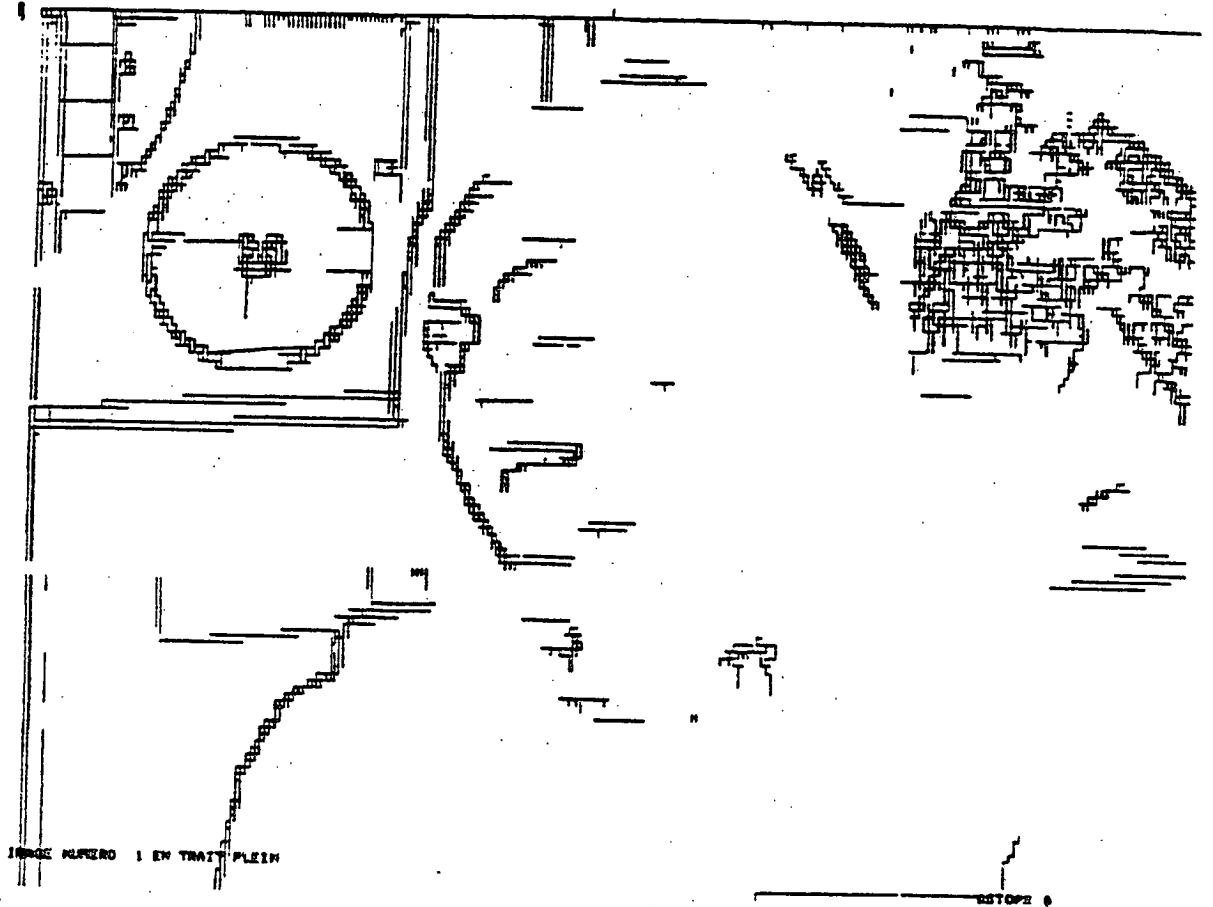


Tab. 2.6

d)



e)



Tab. 2.6

f)



Conclusion

L'algorithme de suivi de contours et le processus de validation associé a permis d'isoler et de caractériser les contours qui seront considérés comme des zones-rupture pour l'estimateur de mouvement.

On remarque qu'effectivement parmi ces contours validés se retrouvent les principales frontières d'objet (susceptibles d'être en mouvement) : comme le bord du visage, l'épaule, la mire etc... mais aussi des gradients de zone "texture". Les ruptures ne pourront donc pas être systématiques.

La partie 4 de ce chapitre visera à suivre temporellement ces contours validés.

En dernier lieu, nous pouvons remarquer que cet algorithme de suivi de contours peut être appliqué sur toute image à deux dimensions extraite du cube d'intensités de la fig. 1.1a par un autre mode de coupe que le mode spatial (1) présenté dans ce chapitre. Dans la partie suivante, les deux algorithmes MC1 et MC1 bis viseront à l'appliquer au mode (2) et (4) respectivement.

II - (3) : ESTIMATION DE MOUVEMENT DE CONTOURS : ALGORITHME MC1 et MC1 bis

Ce paragraphe a pour but d'appliquer l'algorithme de suivi de contours précédemment au mode spatio-temporel (2) (suivi-ligne temporel). |crit

Les contours considérés dans ce paragraphe sont au sens des contours "spatio-temporels" définis en (II (1.3)).

II (3.1) : Application de l'algorithme

L'algorithme reste strictement le même (description par automate d'état) en gardant toutefois toujours à l'esprit les deux natures distinctes de gradients : les EC_v sont des gradients spatiaux

les EC_h sont des gradients temporels

La variable d'état \hat{H} dans ce cas résulte du filtrage des diverses observations locales de mouvement Δ_x le long du contour. Ce filtrage peut se faire de deux manières :

- *filtrage temporel* : il s'agit de la formule classique de filtrage du paramètre \hat{H}_k dans lequel k ne désigne non plus le n° de ligne ou de marche spatiale mais le n° d'image (ou n° de marche temporelle).

$$\hat{H}_t = Y_t H_{t \text{ obs}} + (1 - Y_t) \hat{H}_{t-1} \quad (\text{algorithme MC1})$$

- *filtrage spatial* : ce filtrage repose sur le fait que le mouvement d'un contour d'une ligne à la suivante est fortement corrélé. Le filtrage porte ici sur le n° de marche spatiale.

$$\hat{H}_k^{(t)} = \alpha_k H_{k \text{ obs}}^{(t)} + (1 - \alpha_k) \hat{H}_{k-1}^{(t)}$$

Ce double lissage peut être résumé dans la formule suivante :
(algorithme MC1 bis)

$$\hat{H} \left(\begin{array}{c} k \\ n^{\circ} \text{mar-} \\ \text{che} \end{array}, \begin{array}{c} t \\ n^{\circ} \text{d'i-} \\ \text{mage} \end{array} \right) = \gamma_{k,t} H_{\text{obs}} + (1 - \gamma_{k,t}) \left[\alpha_{k,t} \hat{H}(k, t-1) + (1 - \alpha_{k,t}) \hat{H}(k-1, t) \right]$$

Les gains étudiés sont du type :

Pour $t = 1$: $\alpha_{k,1} = 0$, $\gamma_{k,1} = f(n^{\circ} \text{ marche temporelle})$
 $= 1, 1/2, 1/3, 1/4 \dots$

Pour $t \geq 2$: $\alpha_{1,t} = 1$, $\gamma_{k,t} = 1/2, 1/3, 1/4 \dots$

$\alpha_{i,t} = f(n^{\circ} \text{ marche spatiale})$

Les gains décroissent puis restent constants en fonction de l'ancienneté du contour.

II (3.2) : Autres modifications dues au mode temporel

* *élimination des "chicots" temporels*

Nous éliminons dans l'observation du H_{obs} la présence de chicots horizontaux et cela afin de ne conserver que l'information mouvement Δ_x utile (cf fig. 3.2a)

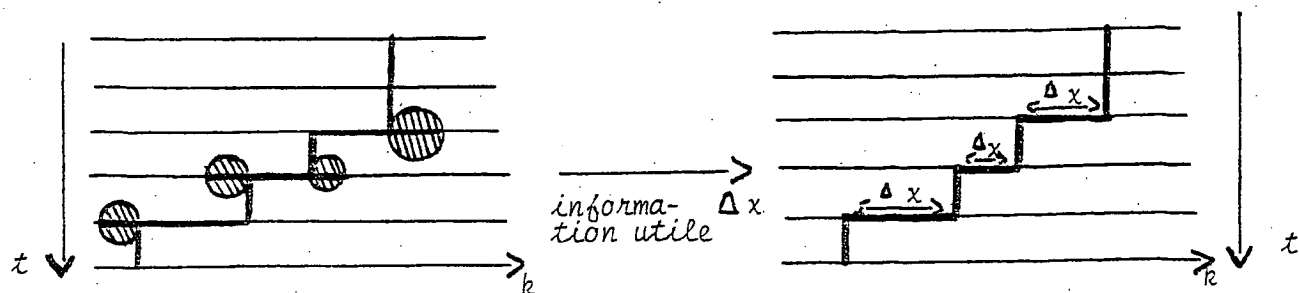
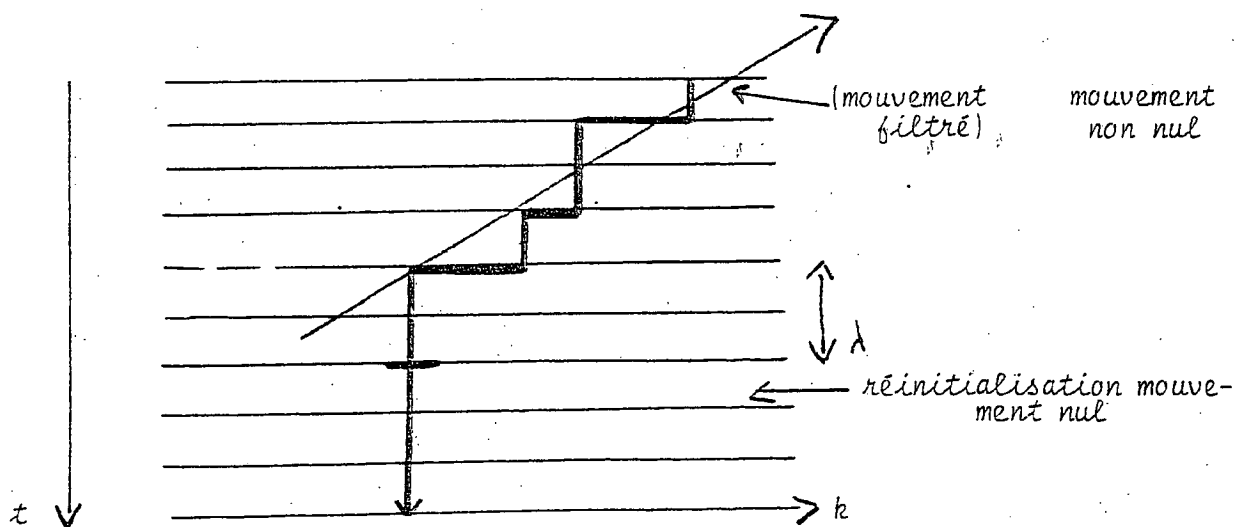


fig. 3.2a

* traitement des contours verticaux temporels :

La présence de tels contours indique la présence d'EC spatiaux fixes. Au lieu d'attendre l'apparition d'un nouveau mouvement non nul (présence d' EC_h), il est possible de réinitialiser l'automate à un état de mouvement nul. Cette réinitialisation se fera avec un retard de lignes ($= 2$) afin de garder un filtrage temporel marche à marche



II (3.3) résultats :

Les résultats portent toujours sur la zone *JOUE* que nous suivons temporellement à travers une séquence de 9 images consécutives (n°31 à 39). L'itération porte cette fois sur le numéro-ligne (*ITERC* = 1)

Le tab. 3.30 : application de l'algorithme avec double lissage pour le paramètre d'état \hat{H}

$$- S_h = S_v = S_t = 10 \quad \lambda \quad r_0 = 2$$

- notations a) "100" point non contour

b) "200" point EC_n : aucune signification au niveau mouvement de contours si ce n'est peut être un déplacement d'un EC_n spatial non détectable par ce mode (2)

c) *sinon* : quantif linéaire de ϕ avec comme précision 1/4 pel/image.

La redondance ligne à ligne est nettement visible puisque pour chaque itération, nous trouvons un "mouvement" d'environ 1/Pel/image en début de séquence et un mouvement nul en fin de séquence.

Tab. 3.31 : suite à cette remarque, seul le lissage spatial est testé. La perte en performance est minime. Par contre la suppression du facteur de lissage temporel supprime la mémorisation nécessaire du mouvement de l'image précédente (mémoire de mouvement $\hat{H}(t-1)$).

Cette approche de filtrage spatial (aussi bien pour le suivi spatial des contours ou pour l'estimation temporelle du mouvement de ceux-ci) permet de penser à inclure l'estimation de mouvement dans l'état interne à l'automate associé au contour spatial. Nous tiendrons compte de cette remarque pour l'algorithme MC2 développé dans la partie qui suit.

tab. 3.32 : grâce à l'estimation précédente du mouvement des contours, nous pouvons effectuer du *recalage de contours* d'une image à l'autre. Seuls sont visualisés ici le recalage des contours validés.

3.32a) affichage des gradients validés de l'image t_0 (trait plein) et de l'image (t_{0+1}) (trait tireté). Le décalage d'un $1/10$ de pel n'est créé que pour les besoins de visualisation.

3.32b) recalage de ces gradients validés par l'estimation de mouvement des contours de l'algorithme MC1. Précision dans le mouvement $1/4$ Pel/image.

3.32c) et d) : mêmes figures pour le couple suivant d'images

Conclusion sur les algorithmes MC1 et MC1 bis

L'application "brute" de l'algorithme de suivi de contours spatial aux modes spatio-temporels (mode (2) et (4)) s'avère être un estimateur de mouvement correct le long des contours validés.

Ces méthodes sont cependant fort lourdes quant au nombre d'automates à gérer (et donc le volume de mémorisation nécessaire) simultanément :

pour l'algorithme MC1 : si n désigne le nombre de connexités spatiales par ligne, L le nombre de lignes d'image, le nombre N d'automates à gérer en parallèle s'élève à :

$$N_{mc1} = n \times L \quad (\text{au lieu de } n)$$

pour l'algorithme MC1 bis : La complexité est encore plus grande (cf Formule double lissage) puisqu'il s'agit de gérer des automates à 2 indices (non indépendants) : t : indice image
 k : indice ligne

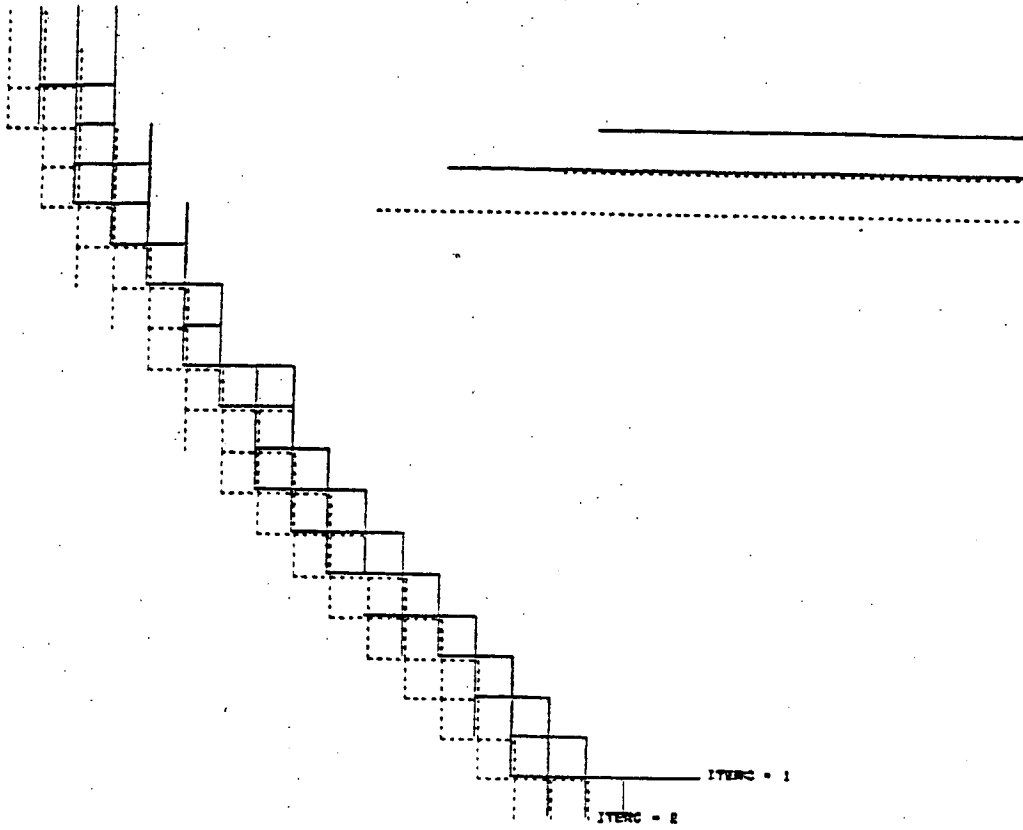
par contre l'étude de ces algorithmes fait ressortir deux points :

. une très forte corrélation ligne à ligne du mouvement des contours validés (pour les gradients non validés cette corrélation est nettement moins élevée)

. un modèle de contour spatio-temporel qui même s'il est observé très localement (sur deux lignes par exemple) est significatif et fournit une information propre du mouvement du contour spatial associé.

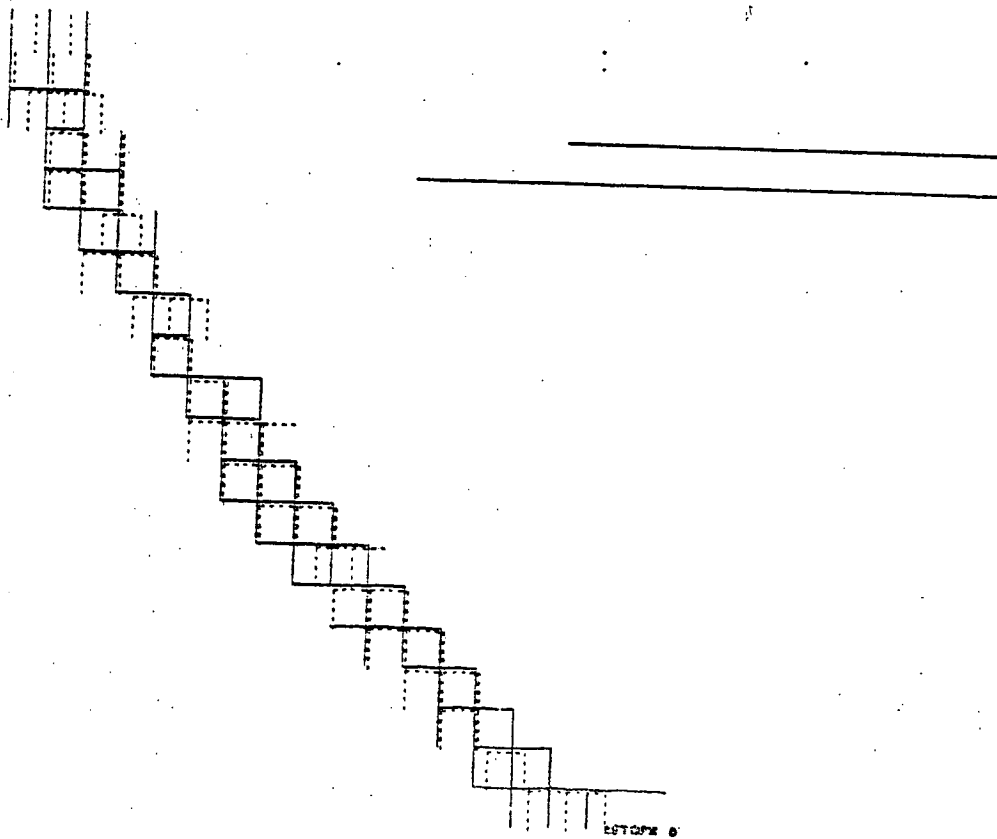
Lissage spatial

a)

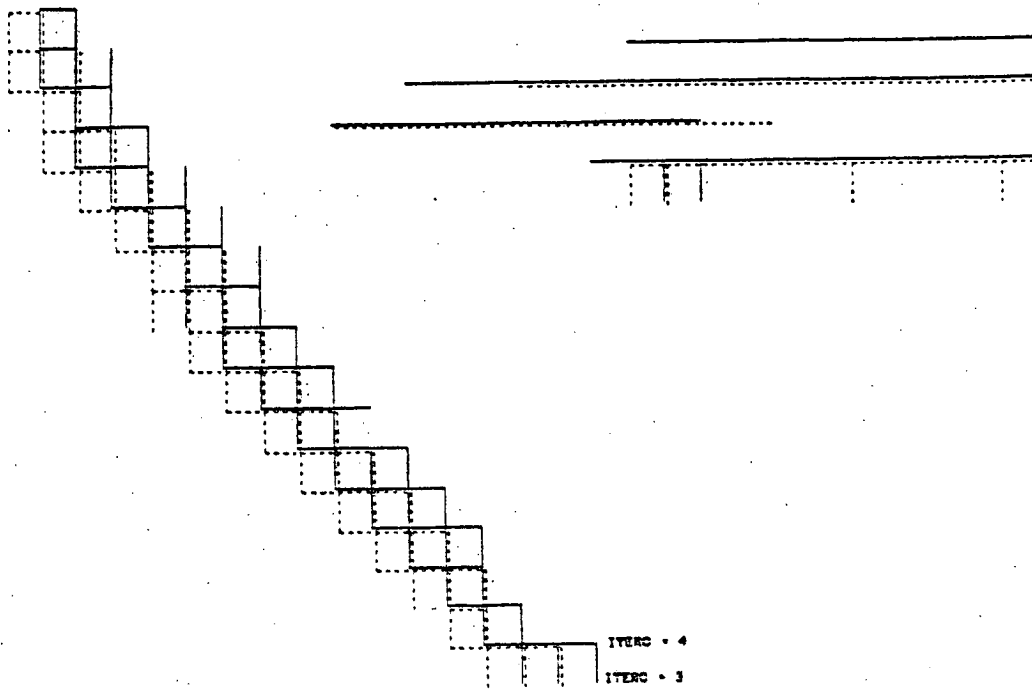


Tab. 3.32

b)

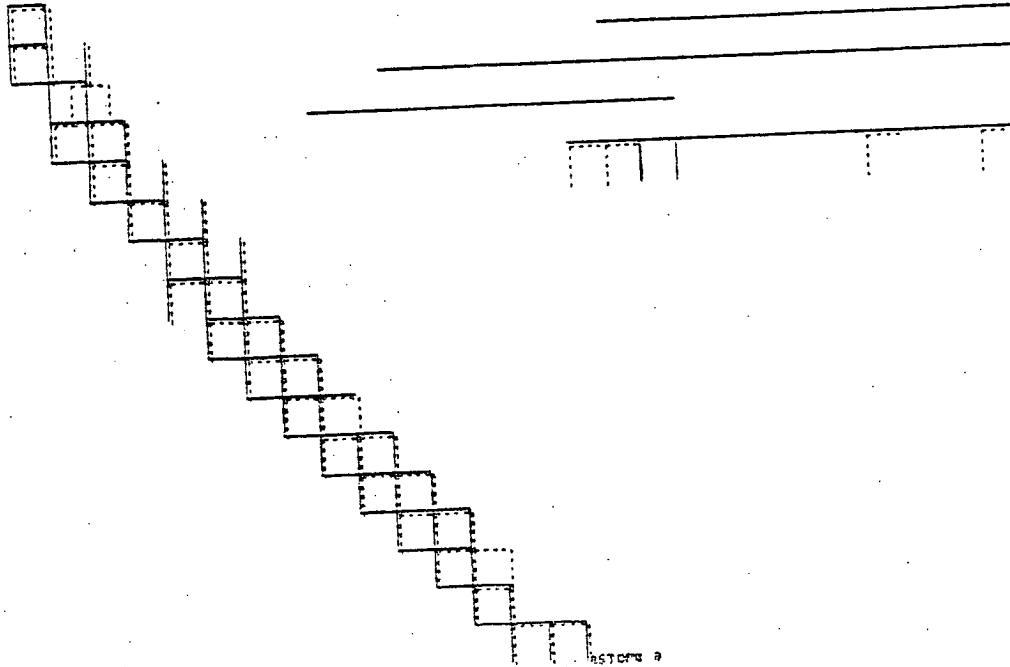


c)



Tab. 3.32

d)



Nous allons utiliser ces deux remarques pour construire un estimateur de mouvement des contours qui soit inclus dans la description par automate du contour spatial.

II - (4) : ESTIMATEUR "SIMPLIFIE" : ALGORITHME MC2

II (4.1) : Observation locale du mouvement :

Cette observation locale va porter sur un couple de lignes d'images (la même ligne dans deux images consécutives)

Notations :

Quatres variables mesurent le déplacement local de la connexité C_t par rapport aux éléments de contour validés (au sens des tests de validation du II.2) appartenant à l'image précédente.

Nous noterons : a) $DFDG$: le déplacement du front droit vers la gauche ($\Delta_x < 0$)

b) $DFGD$: déplacement du front gauche vers la droite

c) $DFDD$, $DFGG$...

Schéma :

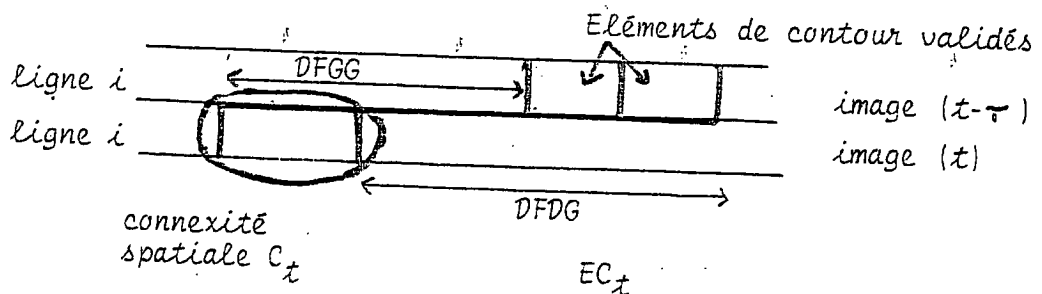


Fig. II (4.1)

L'exemple de la figure 4.1 montre le cas d'une connexité spatiale C_t reliée temporellement à des EC spatiaux et validés : il s'agit dans ce cas d'un contour qui se "déplacerait" localement vers la gauche ($\Delta_x < 0$) puisque $DFDD$ et $DFGD$ sont nuls

Estimation locale :

Elle est définie de la manière suivante :

$$\begin{aligned} \text{soit } DGD & \begin{cases} = DFDG \text{ si non nul} \\ = DFDD \text{ sinon} \end{cases} \\ \text{et } DFG & \begin{cases} = DFGD \text{ si non nul} \\ = DFGG \text{ sinon} \end{cases} \end{aligned}$$

alors l'estimation locale $H_{t \text{ obs}}$ est définie par

$$H_{t \text{ obs}} = \frac{DFD + DFG}{2}$$

Remarques

Les conflits qui peuvent se poser (DFD et DFG de signe contraire par exemple) sont atténués par la moyenne faite dans $H_{t \text{ obs}}$. Pour deux mouvements de même amplitude et de signe opposés, $H_{t \text{ obs}}$ estimé nul. la connexité est considérée comme immobile.

La variation d'épaisseur de la connexité (variation de phase de numérisation sur le contour) est prise en compte dans l'estimation comme un mouvement.

L'introduction de la notion de "fourches temporelles" n'est plus nécessaire ici puisque les EC de l'image $t - \gamma$ reliés temporellement à C_t peuvent ou non appartenir à une même connexité spatiale de l'image $(t - \gamma)$ sans que l'estimation de $H_{t \text{ obs}}$ local en soit affecté. La notion d'automate associé à chaque contour n'intervient pas dans le calcul lui-même de l'observation locale du mouvement.

Cette observation $H_{t \text{ obs}}$ va constituer par contre une nouvelle entrée pour l'automate associé à C_t .

II (4.2) : Modification de l'état interne de l'automate associé à C_t

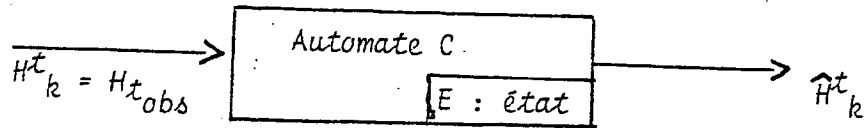


Fig. 4.2a

A l'automate décrit à la figure 2.1a, nous ajoutons l'entrée H_k^t qui estime localement le mouvement de la k ième marche du contour C_t . A cette nouvelle entrée est associée la variable d'état \hat{H}_k^t qui résulte du filtrage suivant :

$$\hat{H}_k^t = \gamma_t^k H_k^t + (1 - \gamma_t^k) \hat{H}_{k-1}^t \quad \text{avec} \quad \gamma_t^k = \frac{1}{\hat{A}_{k-1} + 1}$$

Rupture sur le mouvement : si la nouvelle observation $H_{t_{obs}}^t$ est fortement différente du mouvement filtré \hat{H}_{k-1}^t estimé à la connexion précédente, il y a rupture sur le mouvement et réinitialisation de la variable d'état \hat{H}_k^t .

$$\text{si } \left| H_k^t - \hat{H}_{k-1}^t \right| \geq \lambda_{rt}, \quad \text{alors} \quad \hat{H}_k^t = H_k^t$$

sinon filtrage décrit ci-dessus

Etat interpolé \hat{E} des "fourches spatiales" dans le cas des fourches (cf II 2.1) la variable d'état interpolé \hat{H}_{k-1}^t sera défini par :

$$\text{si } (\hat{H}_k^t \cdot \hat{H}_k^t \geq 0) : \hat{H}_{k-1}^t = \left(\frac{(\hat{N}_{k'}+1) \hat{H}_{k'}^t + (\hat{N}_{k''}+1) \hat{H}_{k''}^t}{(\hat{N}_{k'} + \hat{N}_{k''} + 2)} \right)$$

sinon $\hat{H}_{k-1}^t = \text{indéterminé}$

Remarques :

. \hat{H}_k^t à la différence des autres variables d'état, peut prendre une valeur indéterminée dans le cas où :

- la connexité C_t n'est reliée à aucun élément de contour valide de l'image $(t-\tau)$

- les deux états d'une fourche ont respectivement des mouvements \hat{H}^t de sens différent.

. \hat{H}_k^t peut constituer une sortie de l'automate associé au contour C_t en tant qu'information mouvement sur ce contour.

Il faut rappeler que seule la composante de mouvement horizontale (Δ_x) est prise en compte par cette estimation qui devra donc être corrigée (au sens du paragraphe 1.3) par l'estimation d'orientation $\hat{\theta}_k$

II (4.3) : Utilisation de la procédure de reconstruction des contours validés :

Le calcul de l'observation locale du mouvement dans l'algorithme MC2 (comme le montre la fig. II(4.1)) utilise la validation et la reconstruction des contours validés de l'image précédente. Le but recherché est de ne connecter (par des gradients temporels) la connexité courante de l'image t qu'à de "véritables" connexités-contours de l'image $t-\tau$. On espère ainsi éliminer le recalage temporel des gradients de texture dont on ne peut estimer aucune information locale de mouvement significative.

Les informations de validation utilisées concernent seulement l'image précédente ce qui autorise un retard possible dans le calcul de ces validations de contours.

Résultats :

-- sur la séquence JOUE extraite de la séquence JUDITH

Tab. 4.2a : visualisation de la variable d'état \hat{H}_k^t (précision 1/10) sur les contours validés.

$$\lambda_{h\theta} = 2, \quad \lambda_{ht} = 4, \quad S_h = S_v = S_t = 10$$

$$\gamma_t = 1, 1/2, 1/3, 1/4.$$

Tableau 4.2(b,c,d,e) : recalage des contours validés sur deux couples d'images ((1.2) et (3.4)) grâce à l'estimation de mouvement \hat{H}_k^t

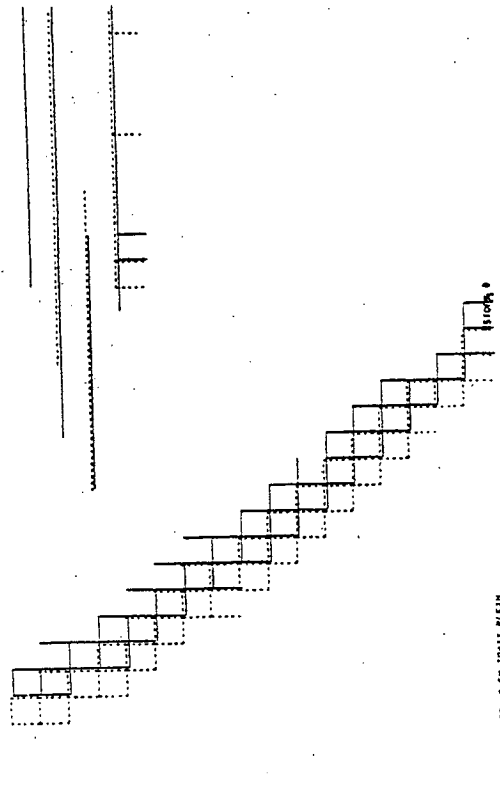
-- sur la séquence BLASON extraite de la séquence CHATEAU

Tableau 4.3a : visualisation de la variable d'état \hat{H}_k^t

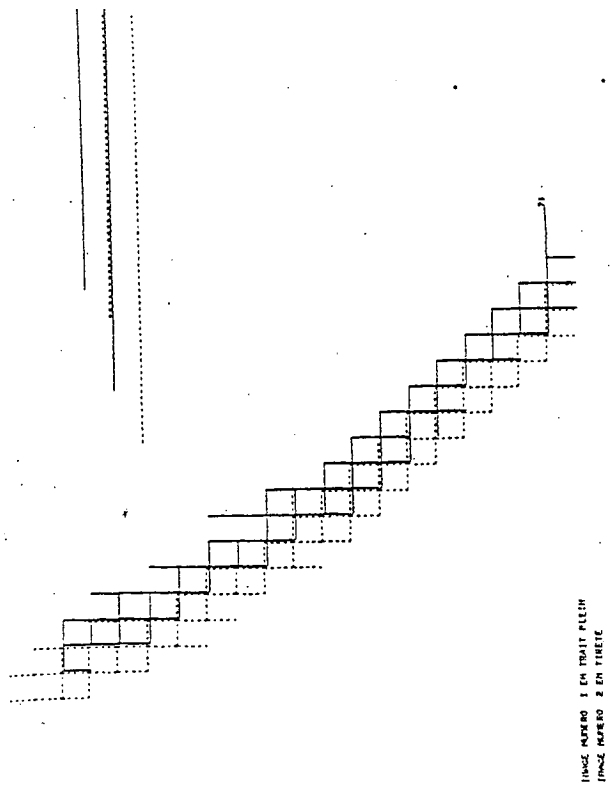
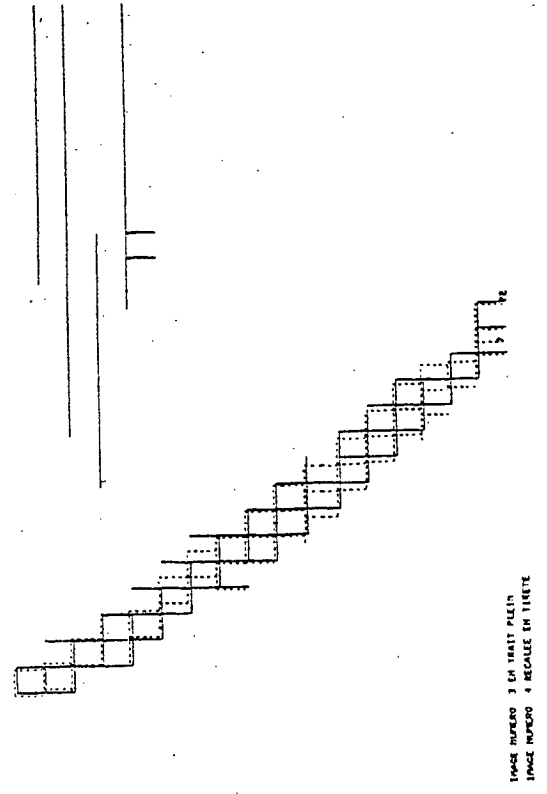
m paramètres qu'en Tab. 4.2a sauf

$$S_h = S_v = S_t = 25$$

Tableau 4.3(b,c) : recalage des contours validés grâce à l'estimation précédente



Tab. 4.2



Tab. 4.2

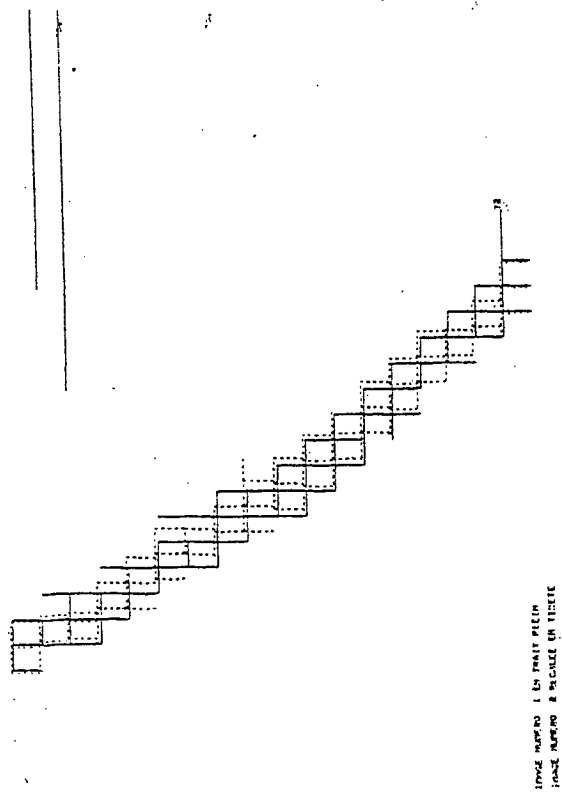
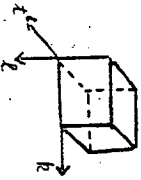
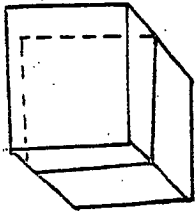
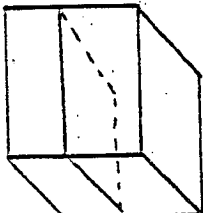
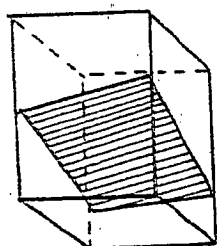
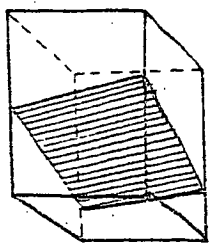


Tableau II 5

Traitement spatial		Traitement spatial + estimation de mouvement		
	Mode Spatial	Algorithme MCI	Algorithme MCI bis	Algorithme MC2
<p>Décomposition du "cube" d'intensités</p> 				
<p>Principe</p> <p>Traitement local</p> <p>+</p> <p>Traitement global</p>	<p>Observation des éléments de contours spatiaux (EC_h et EC_v)</p>	<p>observation d'éléments de contours spatiaux (EC_v) et temporels (EC_t)</p>		
	<p>Automate associé à chaque contour spatial et évoluant ligne à ligne</p>	<p>Automate associé à chaque intersection ligne/contour spatial évoluant image par image</p>		
	<p>nature du liage spatial</p>	<p>temporel</p>	<p>spatial + temporel</p>	
<p>complexité (en nb d'automates à gérer)</p>	<p>$N_{spa} = n$</p>	<p>$N_{MCI} = n \times L$</p>	<p>n automates à 2 indices complexité $> N_{mci}$</p>	
			<p>Automate associé à chaque contour spatial évoluant ligne à ligne</p>	
			<p>spatial</p>	
			<p>$N_{MC2} = n$</p>	

II - (5) : RECAPITULATIF, SOMMAIRE de la METHODE MC2

. Le Tableau II 5 présente un tableau comparatif entre le traitement spatial et les trois traitements temporels envisagés.

Nous précisons pour chaque type de traitement, la nature des automates à définir et quel lissage leur appliquer.

En complexité (c.a.d. en nombre d'automates à gérer) , l'algorithme MC2 est avantageux puisqu'il est identique au traitement spatial bien que prenant en compte l'estimation de mouvement.

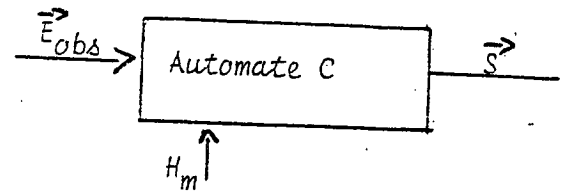
Nous résumons ici la version (MC2) adoptée dans la suite de l'étude quant à la modélisation par automate des contours :

1) A chaque contour est associé un *automate d'état C* ayant comme :

* observations d'entrée \vec{E}_{obs}

* estimations de sortie \vec{S}

* horloge : horloge-marche



2) le vecteur d'entrée \vec{E}_{obs} possède 5 composantes :

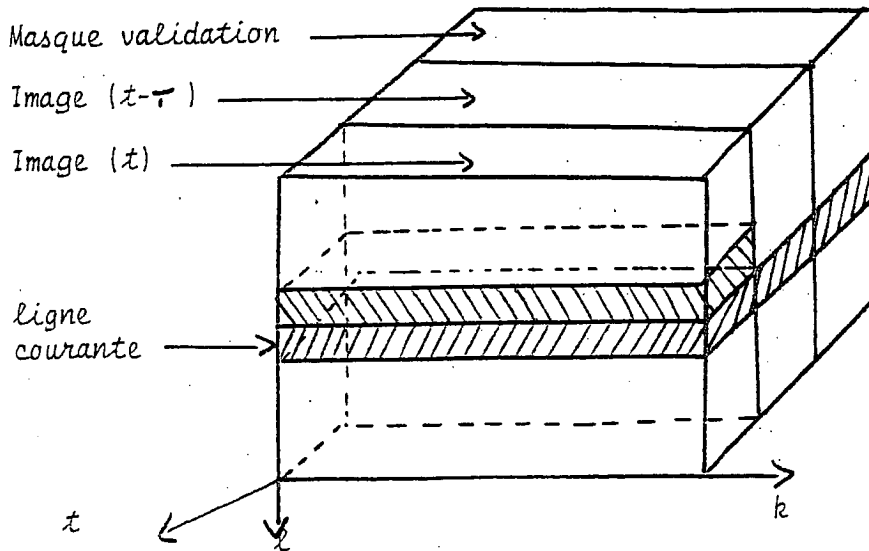
$$\vec{E}_{obs} = (H_k, V_k, P_k, S_k, H_k^t)$$

Ces mesures sont observées par calcul de gradients spatiaux et temporel et détection d'éléments de contour EC_h, EC_v, EC_t

Dans le "cube" d'intensités initial (cf II 1.) la fenêtre d'observation pour chaque ligne se résume donc à :

. la ligne précédente de la même image (\rightarrow détection EC_h)

. la même ligne de l'image précédente + son masque de validation des éléments de contours présents dans cette ligne (\rightarrow détection EC_t)



Fenêtre d'observation

3) l'initialisation de l'état E interne de l'automate se fait avec les valeurs du vecteur d'observations \vec{E}_{obs}

4) la réactualisation de l'état se fait par filtrage récursif au fil des marches, des observations reçues.

$$\vec{E} = (\hat{H}_k, \hat{V}_k, \hat{F}_k, \hat{Q}_k, \hat{N}_k, \hat{A}_k, \hat{H}_k^t)$$

5) le processus de validation (définition du masque des EC_y) consiste en des tests partant sur les paramètres d'état.

6) le vecteur \vec{S} sortie de l'automate produit trois estimations :

$$\vec{S} = (\hat{\theta}_k, \hat{sign}_k, \hat{H}_k^t)$$

respectivement estimation d'orientation de signe et de mouvement des contours présents à la marche n°k

7) deux cas de ruptures dans le processus de réactualisation de E peuvent se produire :

* rupture de modèle : du à l'orientation $\theta_{obs} \neq \hat{\theta}_{k-1}$

au signe $\hat{q}_k \geq 2$

* rupture de mouvement : $H_{obs}^t \neq \hat{H}_{k-1}^t$

Le processus de réinitialisation peut porter suivant le type de rupture sur la totalité ou une partie des composantes du vecteur d'état.

3eme PARTIE

ESTIMATION DU MOUVEMENT EN TOUT POINT D'IMAGE

III - (1) : OBJECTIFS, TYPE D'ALGORITHMES VISES

III (1.1) : Objectifs

* Nous avons estimé, dans le chapitre précédent, une information mouvement sur des points d'image bien précis, ceux qui appartenant à une connexité spatiale d'un contour, étaient reliés temporellement à des éléments de contours validés de l'image précédente. Le but de cette troisième partie est de construire un estimateur de mouvement, cette fois en tout point d'image. L'approche adoptée pour l'estimation de mouvement est donc une approche locale point à point à la différence de travaux tels que [14 à 18] qui associent à un objet préalablement détecté, un vecteur déplacement, ou qui opèrent sur un bloc de points d'image.

Cette approche locale est bien adaptée :

- pour l'optique codage d'images avec compensation de mouvement
- pour le traitement de tout mouvement, sans exception et sans interprétation globale comme il est nécessaire en reconnaissance des formes par exemple. Tout mouvement (ou "changement" de luminance) sera estimé par un vecteur de translation \vec{D} local.

A tout point courant x_i sera associé un vecteur déplacement \vec{D}_{x_i} (noté \vec{D}_i par la suite).

Pour estimer le mouvement en tout point, nous utilisons selon la nature du point courant, les informations suivantes :

- si le point courant est un point contour, l'automate associé au contour nous donne une information à priori sur le mouvement,
- si le point courant n'est pas considéré comme point rupture de mouvement, nous utilisons l'information-mouvement du point précédent (le terme "précédent" étant pris au sens du processus balayage-ligne de la télévision).

III (1.2) : Type d'algorithmes visés

La propagation point à point de l'estimation de mouvement va utiliser un algorithme adaptatif à gain constant du type :

$$\hat{D}_i = \hat{D}_{i-1} - \epsilon_f(\hat{D}_{i-1}, x_{i-1}, t)$$

où f désigne la fonction de correction minimisant un critère donné.

Comme précisé dans le résumé bibliographique, le choix du gain ϵ résulte d'un compromis entre la vitesse de convergence souhaitée et le bruit d'agitation résiduelle.

L'application de tels algorithmes a déjà été menée par le Pr *Netravali* et son équipe, sur des séquences visiophones noir et blanc et couleurs, comportant des petits mouvements et des variations d'illumination.

III - (2) : COMPARAISON de DIVERSES PROPAGATIONS SANS RUPTURE

III (2.1) : Les diverses propagations choisies tentent de minimiser récursivement $\mathbb{E} \left(DFD(x, \hat{D}_i)^2 \right)$ ou \mathbb{E} est à interpréter comme une moyenne spatiale dans le voisinage du point considéré.

Avec les notations choisies en 1ère partie :

$$DFD(x_i, \hat{D}_{i-1}) = I(x_i, t) - I(x_i - \hat{D}_{i-1}, t - \tau)$$

$$\begin{aligned} * \text{ Or } \nabla_D \left(DFD(x_i, \hat{D}_{i-1})^2 \right) &= 2 \times DFD(x_i, \hat{D}_{i-1}) \times \left(\nabla_D (DFD(x_i, \hat{D}_{i-1})) \right) \\ &= 2 DFD(x_i, \hat{D}_{i-1}) \nabla I(x_i - \hat{D}_{i-1}, t - \tau) \end{aligned}$$

avec $\nabla I(x)$ désignant le vecteur gradient spatial d'intensité au point d'image x

* l'algorithme adaptatif de propagation de l'estimateur est donc défini par :

$$\begin{aligned} \hat{D}_i &= \hat{D}_{i-1} - (\xi / 2) \times \nabla_D \left(DFD(x_i, \hat{D}_{i-1})^2 \right) \\ &= \hat{D}_{i-1} - \xi DFD(x_i, \hat{D}_{i-1}) \nabla I(x_i - \hat{D}_{i-1}, t - \tau) * \end{aligned}$$

La fonction de correction apporte une modification sensible l'estimation du mouvement du point précédent par :

- . les points à forte erreur temporelle (DFD)
- . les points à fort gradient spatial (∇I)

* N.B. : Le calcul de ces deux termes (DFD et ∇I) nécessitent des interpolations (cf remarque en fin de partie (3)). Pour l'instant nous prenons des interpolations linéaires avec maximum de précision (c.a.d sans troncature des \hat{D})

III (2.2) : Résultats (zone JOUE extraite de la séquence Judith cF fig. II 2.2 a,c,d)

Pour les comparaisons présentées ci-après, nous utiliserons :

* des images d'erreurs de prédiction à trois niveaux de quantification :

$$e = I(x,t) - P_{x,t}$$

$I(x,t)$: luminance (8bits)
 $P_{x,t}$: prédiction

niveau noir : $|e| \geq 15$

niveau gris : $6 \leq |e| < 15$

niveau blanc : $|e| < 6$

* des histogrammes d'erreurs de prédiction comprises entre - 25 et + 25.

. Tableau 3.1 : différences interimages (c'est à dire $P_{x,t} = I(x, t-\tau)$ en tout point x de la zone Joue à l'image (t) et $(t+1)$)

Cette figure confirme l'hypothèse de base selon laquelle les zones critiques à erreurs temporelles élevées (donc susceptibles d'être des zones-ruptures de mouvement) se trouvent principalement sur les contours (ici le contour de la joue). En zone texture, les erreurs temporelles moins franches proviennent à la fois du mouvement de la texture et de l'enchevêtrement temporel des gradients de texture qui en résulte.

La dissymétrie de l'histogramme provient des déplacements quasi-unidirectionnels présents dans la scène (mouvements de droite à gauche cF)

. Tableau 3.2 : différences déplacées (DFD) (c.a.d. $P_{x,t} = I(\hat{x}-\hat{D}_{x-1}, t-\tau)$)

La propagation choisie ici est la propagation avec fonction de correction "complète" : $f = DFD \times \nabla I$. Elle sera notée par la suite propagation PSRO. $\xi = 1/1024$

La réinitialisation à zéro de l'estimateur est effectuée en chaque début de ligne.

On observe qu'une simple propagation par algorithme adaptatif ne résoud pas les ruptures possibles de mouvement sur contours (erreurs élevées tout au long du contour). En zone texture l'amélioration est sensible mais des divergences importantes de l'estimateur peuvent y apparaître lorsque celui-ci se trouve en situation de mauvais accrochage.

. Tableau 3.3 : différences déplacées (DFD)

La propagation choisie est une propagation "simplifiée"

$$\widehat{D}_i = \widehat{D}_{i-1} - \varepsilon \operatorname{sign} \left(DFD(x_i, \widehat{D}_{i-1}) \right) \operatorname{sign} \left(\nabla I(x_i - \widehat{D}_{i-1}, t - \tau) \right)$$

Nous la noterons par la suite *PSR1* $\varepsilon = 1/8$

Les seules corrections de \widehat{D}_{i-1} autorisées avec *PSR1* sont $+\varepsilon$ ou $-\varepsilon$ ce qui constitue un lissage des divers sauts d'estimation (élimination des sauts trop brusques dus par exemple au bruit de texture). L'inconvénient est de limiter la vitesse de convergence (fixée à ε) même en cas de point-rupture de mouvement (augmentation des dégradations des erreurs temporelles présentes sur le contour par rapport au tableau 3.2)

III - (3) : PROPAGATION avec RUPTURE de MOUVEMENT

L'idée de base est d'utiliser l'information-mouvement sur les contours calculée au chapitre II.

III (3.1) : Réinitialisation systématique de l'estimateur

Si au point courant, on connaît (au sens de l'algorithme MC2) une information-mouvement à priori sur ce point, la propagation récursive le long de la ligne est interrompue, et l'estimateur est réinitialisé brutalement à la valeur :

$$\vec{D}_1 = \begin{pmatrix} D_{obs} \times \cos^2 \theta \\ -\sin \theta \cos \theta D_{obs} \end{pmatrix}$$

où * D_{obs} désigne l'information-mouvement calculée sur la rupture par MC2

* θ désigne l'orientation du contour (sortie de l'automate)

Les algorithmes de propagation définis précédemment étant des

propagations bidimensionnelles, la réinitialisation par \vec{D}_1 au lieu de $(D_{obs}, 0)$ compense le caractère non isotrope de l'estimation de mouvement de l'algorithme MC2.

. *Tableau 3.4* : Images des différences déplacées (DFD) avec :

- rupture le long du contour (sauf aux quatre premières lignes puisque le contour est trop jeune et les règles de validité ne sont pas vérifiées).

- fonction de correction nulle ($\hat{D}_i = \hat{D}_{i-1}$)

. *Tableau 3.5* : idem que Tab.3.4 mais avec une fonction de correction non nulle : $f = \xi \times \text{sign DFD} \times \nabla I$ $\xi = 1/8$

La propagation employée ici sera appelée propagation PR1

. *Tableau 3.6* : gain "optimum" $\xi = 0.15$

L'ajout, à l'estimation récursive, d'un détecteur de rupture instantané permet de cumuler les deux effets suivants :

- * suppression des fortes erreurs temporelles dues au mouvement des contours (frontières d'objets par exemple, la joue dans le cas présent)

- * affinement local de l'estimation et suppression des erreurs temporelles dues au "bougé" des gradients de texture (la bouche ici).

III (3.2) : Tests de rupture

Le cas de la zone JOUE est une situation simple où le contour de la joue sépare deux zones bien distinctes à mouvements différents. Des situations beaucoup plus sévères se présentent, notamment lors de contours internes à un objet en mouvement ou lors de croisement d'objets en mouvement. Se pose alors le problème d'affectation de zones d'images aux diverses informations-mouvement dont on dispose. La réinitialisation de l'estimateur lors du franchissement d'un contour-rupture dépend de la réalisation de plusieurs tests.

* Définition des tests de rupture : les différentes informations de mouvement utilisées dans ces tests sont résumées à la fig.3.b

- \vec{D}_1 ** : information de mouvement à priori sur la connexité rupture courante.

- \vec{D}_g * : mouvement de la "frontière" gauche de l'objet estimé à la ligne précédente (au point n° n_g)

- $\vec{D}_{d\perp}^{**}$: information à priori sur le mouvement de la "frontière" droite
- \vec{D}_m^* : mouvement "median" entre les deux frontières (point n_m)

Cette information-mouvement a été choisie pour permettre à l'estimateur récursif de converger en zone non contour.

Le point n_m peut être choisi à partir du bord gauche ($n_m = n_g + r$) ou du bord droit ($n_m = n_d - r$).

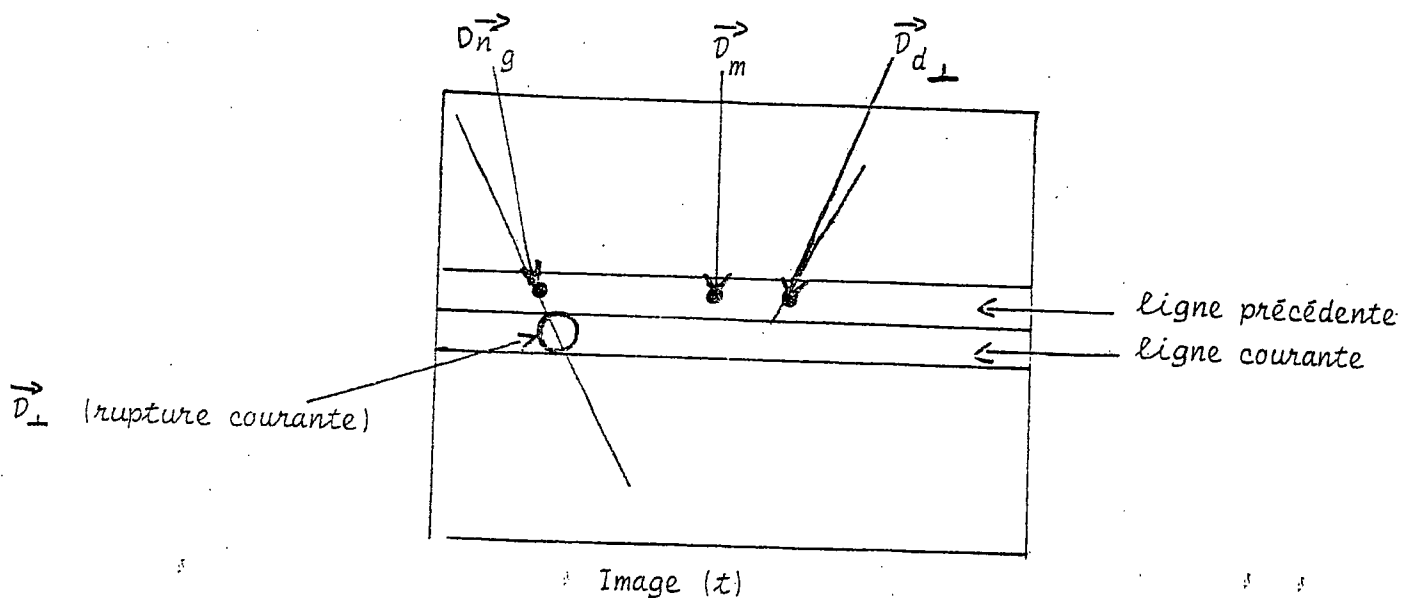


Fig.3.b

Ces quatre informations vont permettre de définir la valeur de réinitialisation $\vec{D}_{réinit.} = \vec{D}_{x_c}$ de l'estimateur au point-rupture courant

Une information supplémentaire intéressante pourrait être ajoutée : celle du mouvement du point précédent $\vec{D}_{x_c - 1}$

* les informations \vec{D}_g et \vec{D}_m sont les estimations finales (de l'algorithme complet d'estimation) de mouvement aux points n_g et n_m .

** les informations \vec{D}_{\perp} et $\vec{D}_{d\perp}$ sont les estimations brutes du mouvement sur les contours (calculées par l'algo MC2)

Test T1 : estimation de $\hat{\vec{D}}$ réinit.

$$\text{si } |\vec{D}_d - \vec{D}_m| > \lambda_1 \quad \hat{\vec{D}}_{\text{réinit.}} = x_1 \vec{D}_g + (1-x_1) \vec{D}_m$$

(cas de conflits de mouvement entre n_m et n_g)

$$\text{si } |\vec{D}_d - \vec{D}_m| \leq \lambda_1 \quad \hat{\vec{D}}_{\text{réinit.}} = x_1 \vec{D}_m + (1-x_1) \vec{D}_d$$

Test T2 : calcul de la valeur de réinitialisation

$$\text{si } |\vec{D}_d - \hat{\vec{D}}_{\text{réinit.}}| > \lambda_2 \quad \vec{D}_{\text{réinit.}} = \hat{\vec{D}}_{\text{réinit.}}$$

$$\text{si } |\vec{D}_d - \hat{\vec{D}}_{\text{réinit.}}| \leq \lambda_2 \quad \vec{D}_{\text{réinit.}} = x_2 \hat{\vec{D}}_{\text{réinit.}} + (1-x_2) \vec{D}_d$$

* Résultats : (Tab.3.7,3.8 : rappel sur l'estimation MC2 sur BLASON)

. Tableau 3.7 : la zone BLASON (plusieurs bandes de luminance différente) est extraite de la séquence CHATEAU. Cette zone-test a été sélectionnée pour son mouvement rapide (3 à 5 pels/image), ses nombreuses zones-ruptures et l'enchevêtrement de celles-ci

- tableau 3.7a : visualisation des contours validés à l'image 1 (trait plein) et l'image 2 (tirets)

- tableau 3.7b : recalage des contours par l'algorithme d'estimation de mouvement des contours MC2.

. Tableau 3.8 : résultat de l'estimation de mouvement MC2 (en pel/image) sur la zone BLASON

. Tableau 3.9 : comparaison d'estimation de mouvement (histogrammes à abscisse logarithmique.

a) différences temporelles

b) propagation PSRO (avec la fonction de correction $f = DFDx \nabla I$)

Les mauvaises performances de ce type d'estimateur sont dues aux brusques sauts de gradients et de mouvement présents dans la scène créant de fortes corrections d'estimation et de nombreuses oscillations de l'estimateur.

- c) propagation PSR1 : fonction de correction "*simplifiée*"
- d) algorithme complet : ruptures + f "*simplifiée*" + tests de ruptures

Les tests de rupture ont pour seuils et paramètres :

- $n_m = n_d - 4$
- $x_1 = x_2 = 1/2$
- $\lambda_1 = \lambda_2 = 1$

. Tableau 3.10 : choix du paramètre n_m

- | | |
|--------------------|---------------------|
| a) $n_m = n_g + 4$ | b) $n_m = n_g + 10$ |
| c) $n_m = n_d - 4$ | d) $n_m = n_g + 15$ |

Le cas c le plus favorable montre que l'information médiane de mouvement \hat{D}_m est plus efficace lorsque l'estimateur adaptatif a eu le temps de converger.

Remarques :

L'ensemble des résultats présentés ici a tenu aucun compte de la quantification des erreurs avant transmission sur le canal (cela nécessite un travail d'optimisation du quantificateur travail non abordé dans le cadre de ce mémoire). Il est intéressant à titre comparatif de visualiser l'histogramme des erreurs quantifiées :

- . Le tableau 3.11 : présente de tels histogrammes avec
 - en a) : les différences temporelles
 - en b) : propagation PSR 1
 - en c) : algorithme complet : ruptures + propagation "*simplifié*" + tests de rupture

La quantification choisie est une quantification non uniforme optimisée dans le cadre d'un codage différentiel interimage [31]

<i>n° plage</i>	1	2	3	4	5	6	7	8	9
<i>erreurs</i>	5	5	11	20	34	53	74	99	126
		<i>à</i>	<i>à</i>	<i>à</i>	<i>à</i>	<i>à</i>	<i>à</i>	<i>à</i>	<i>à</i>
		10	19	33	52	73	98	125	255

Dans le calcul de la fonction de correction (simplifiée ou complète) interviennent des interpolations au niveau :

- des calculs des "DFD" (calcul de $I(x-\hat{D}, t-\hat{\tau})$)
- du calcul des gradients spatiaux ∇I aux points $(x-\hat{D})$

L'ensemble des résultats présentés jusqu'à présent utilisait :
 . une précision infinie (du moins celle de l'IRIS 80 sur les réels !) pour l'estimation du vecteur \vec{D}

. une interpolation linéaire exacte à 2 dimensions comme celle décrite dans [20], chère en calculs.

Nous comparons dans le Tableau 3.12 les résultats obtenus dans les trois cas suivants

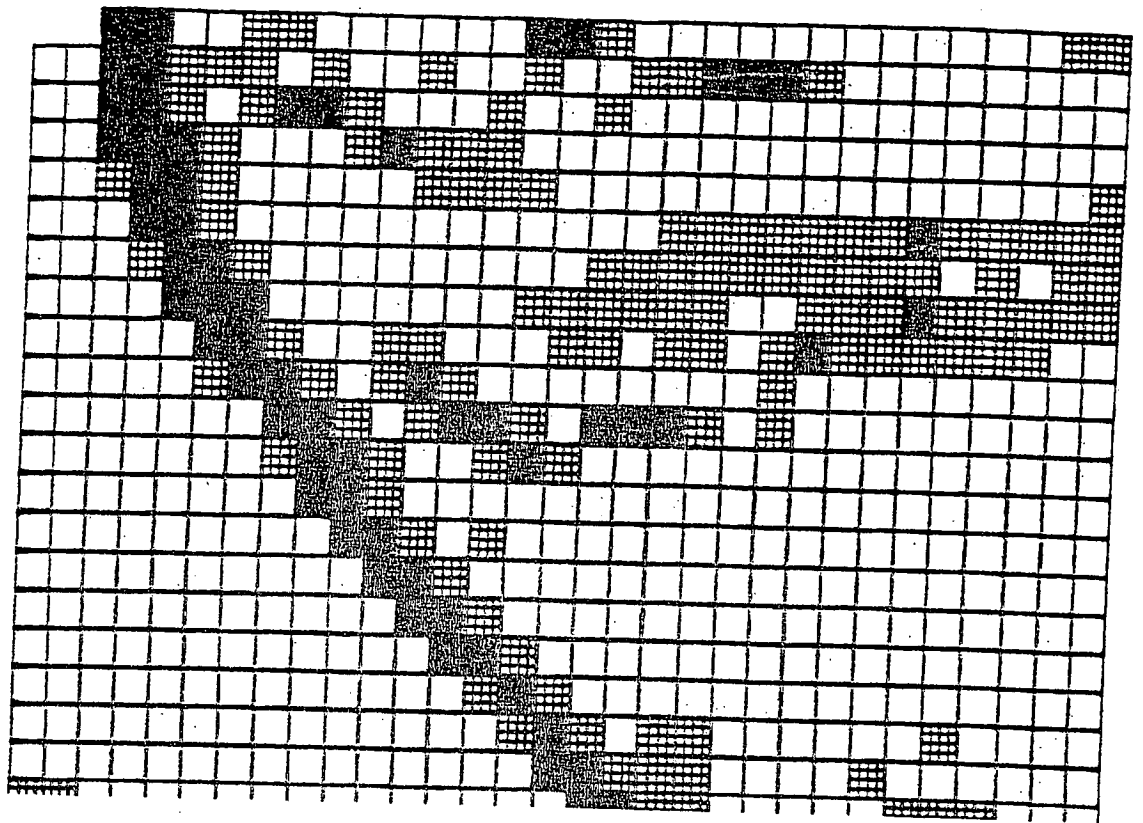
- 3.12 a : - précision "infinie" pour \vec{D}
- calcul d'interpolation exact

(résultat du tableau 3.9d, répété ici comme base de comparaison)

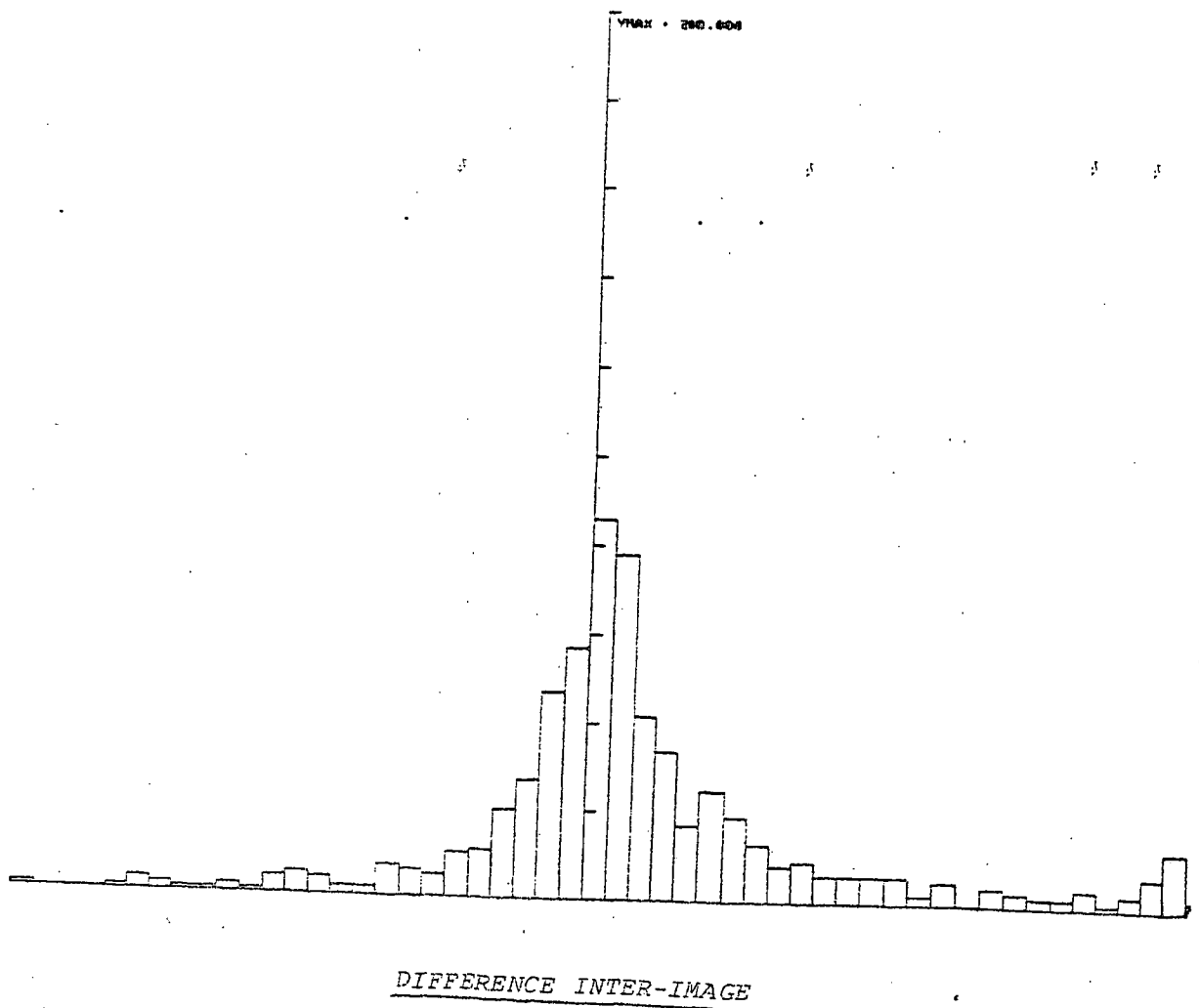
- 3.12 b : - précision à 1/8 pel/image pour l'estimation de \vec{D}
- calcul d'interpolation d'exact

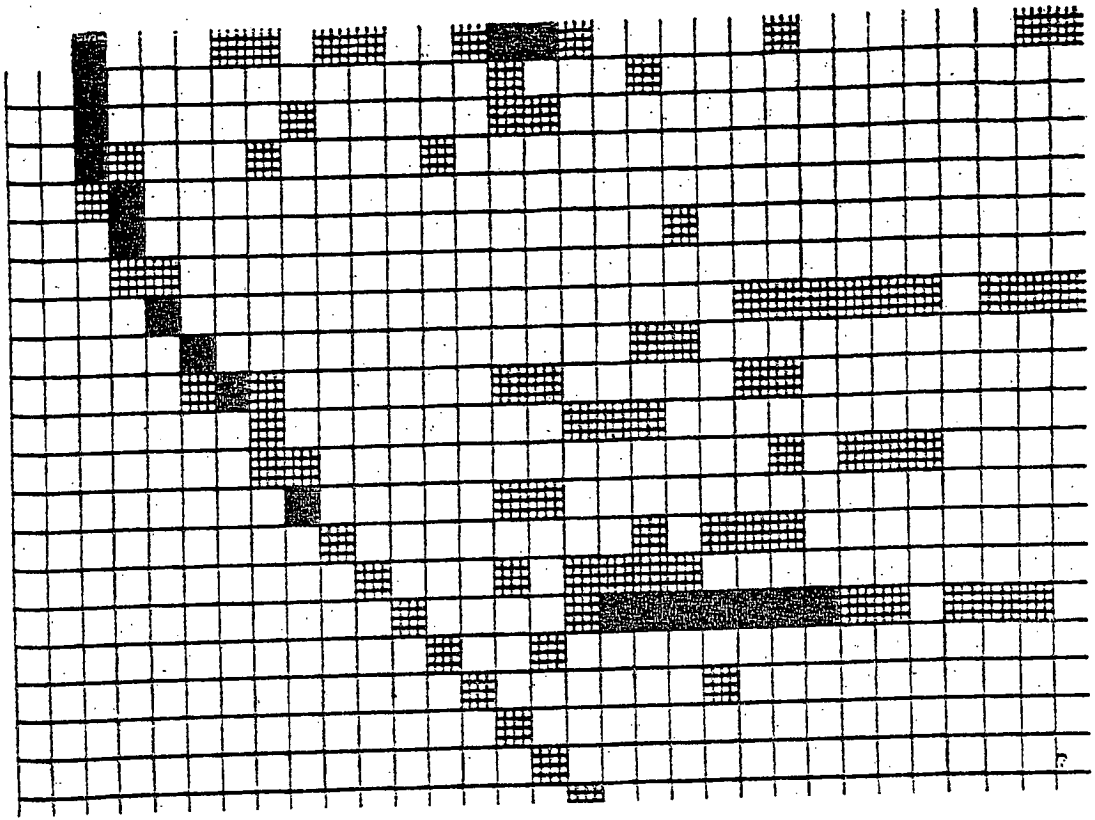
- 3.12 c : - précision "infinie" sur \vec{D}
- grille d'interpolation finie (grille à 1/8 point)

Cette comparaison montre que la précision dans l'estimation du mouvement est plus importante que celle dans le calcul des interpolations

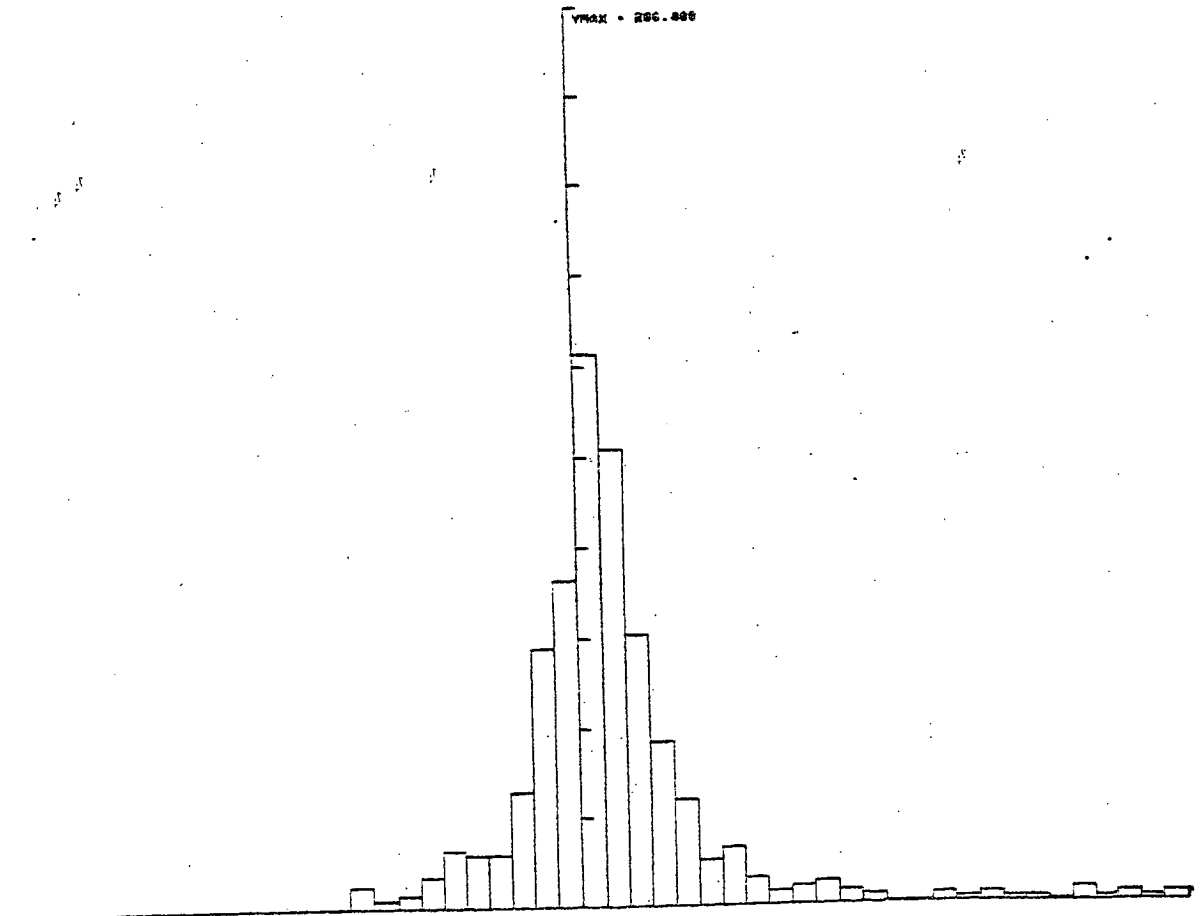


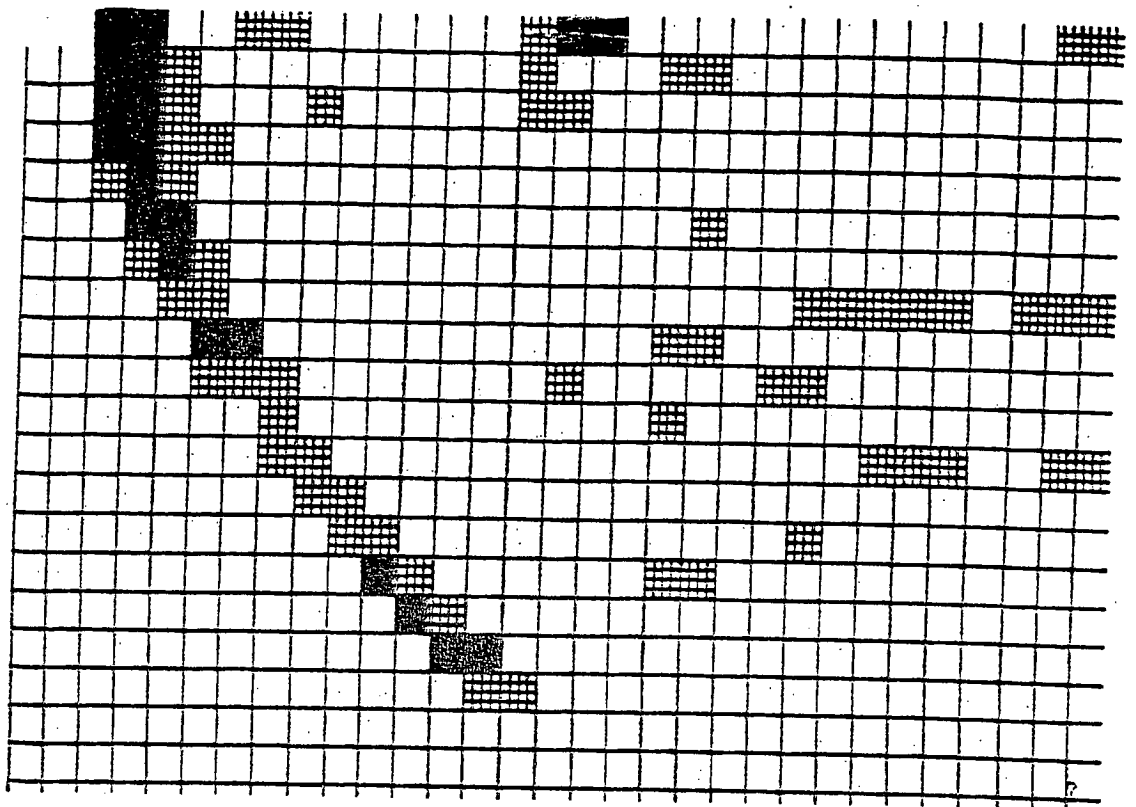
Tab. 3.1



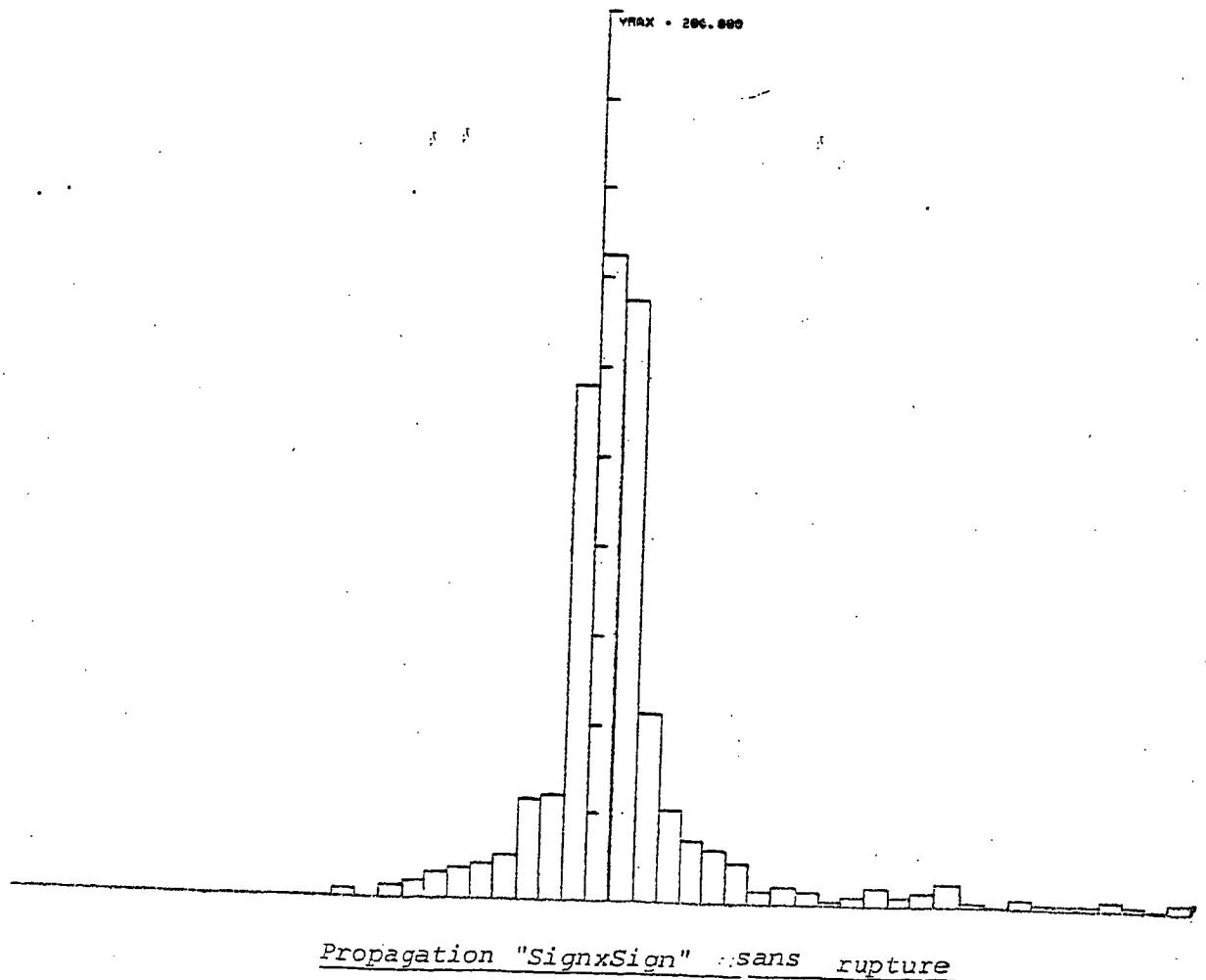


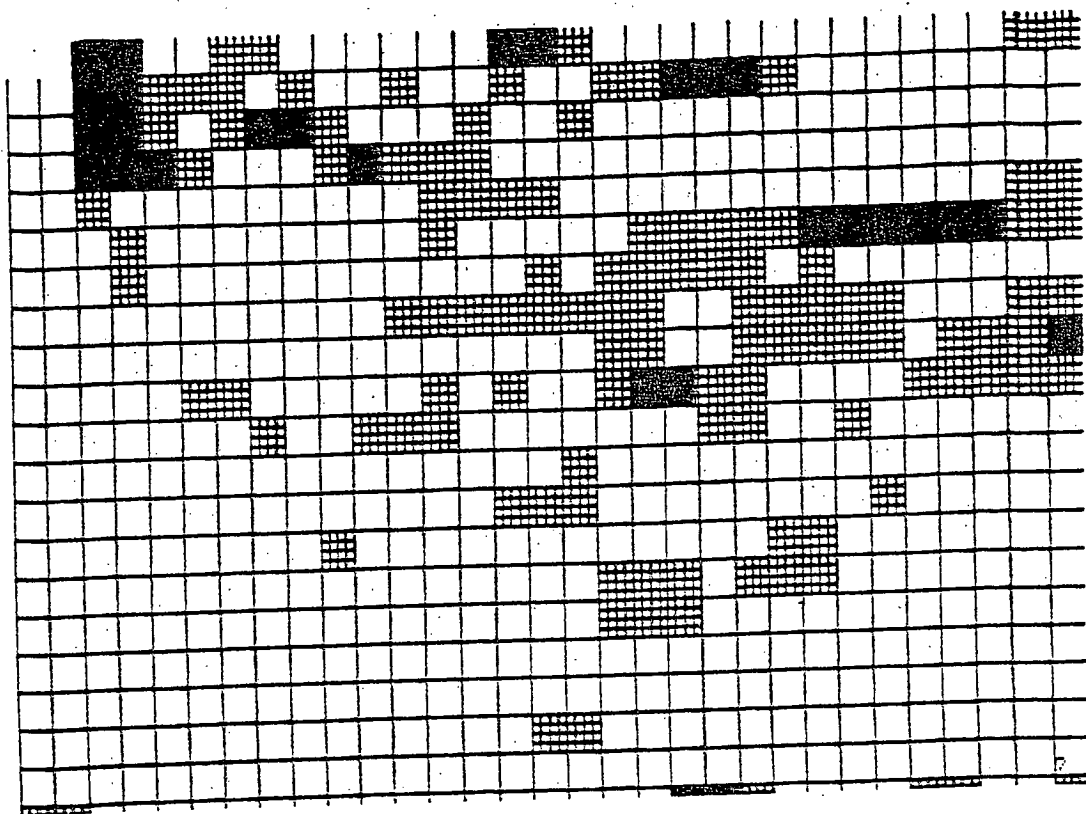
Tab. 3.2



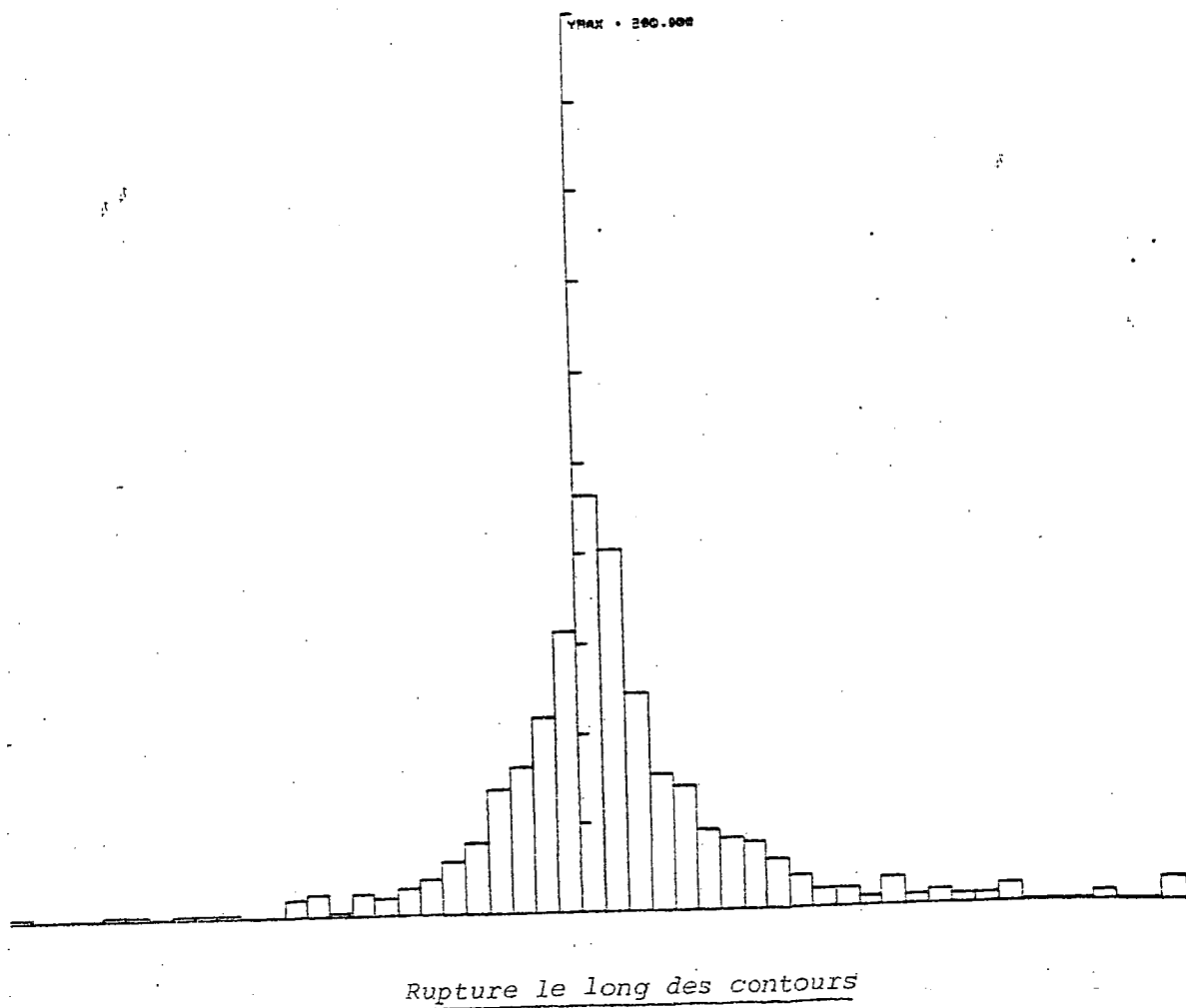


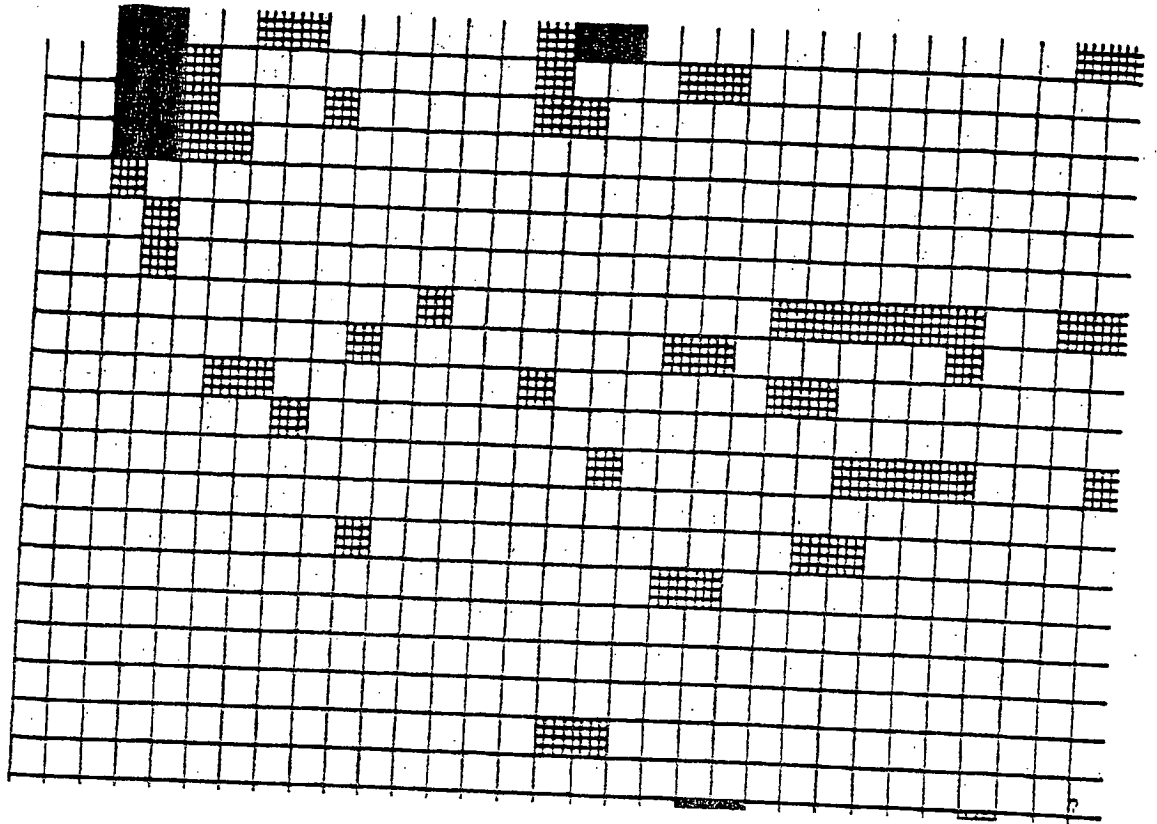
Tab. 3.3



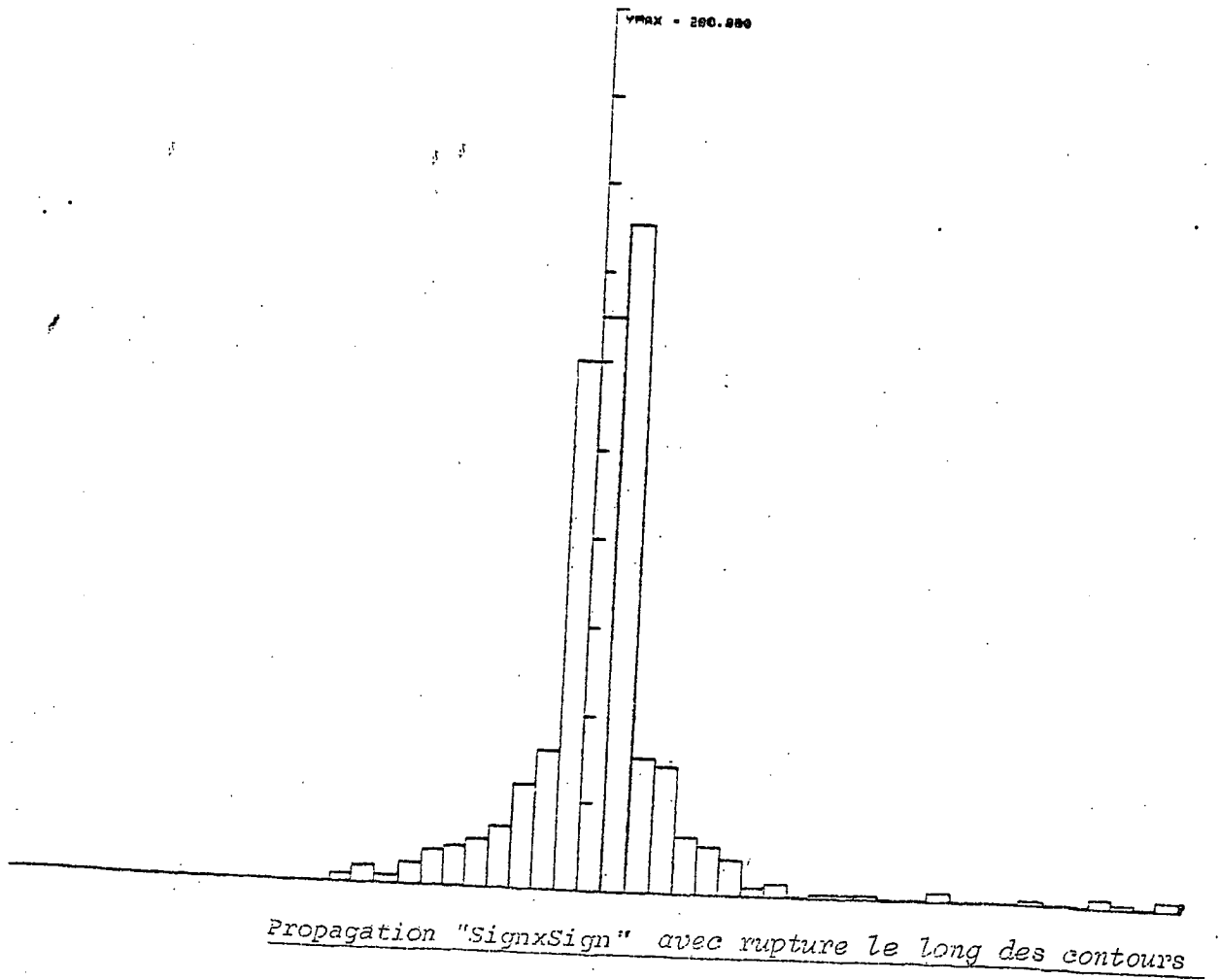


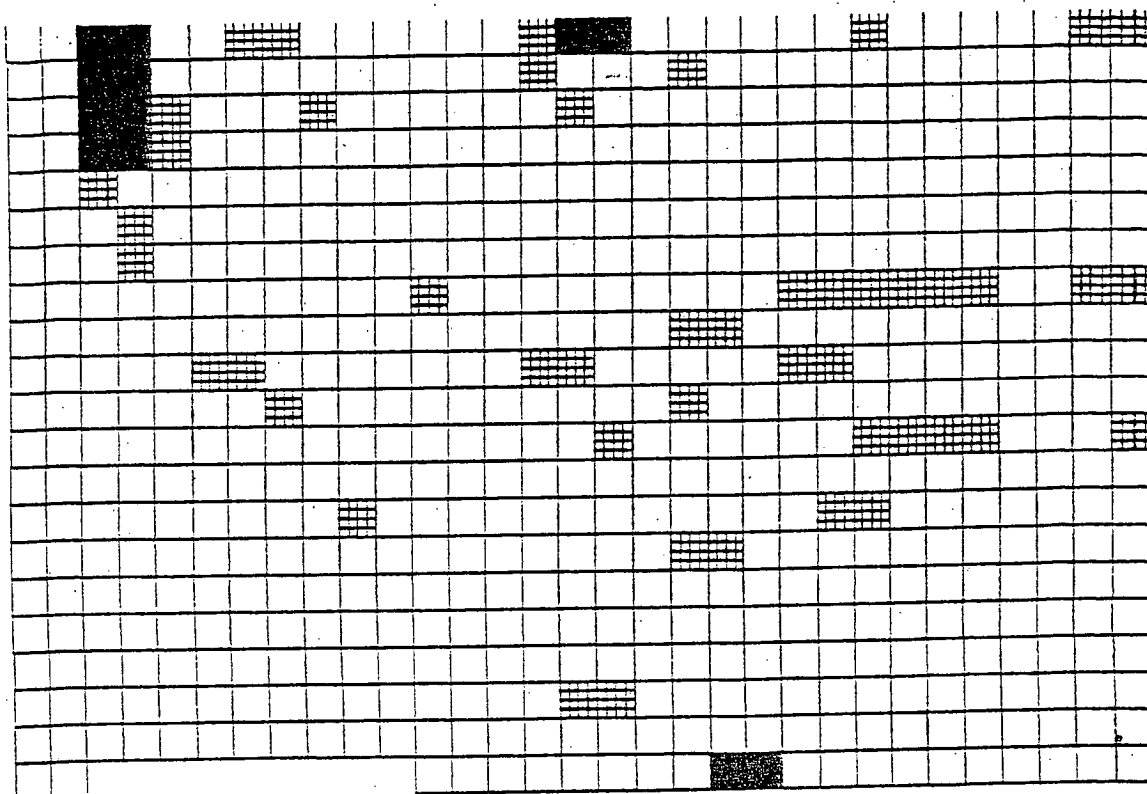
Tab. 3.4



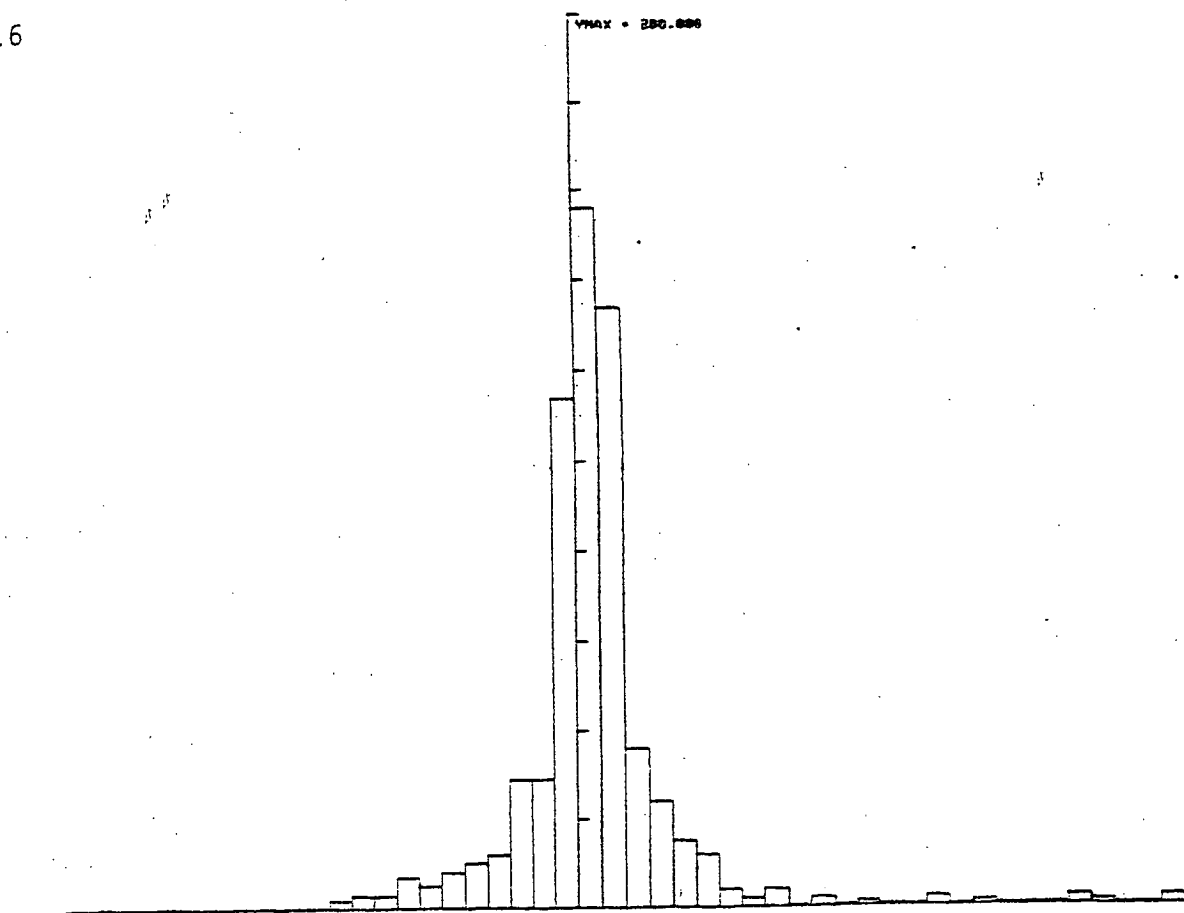


Tab. 3.5



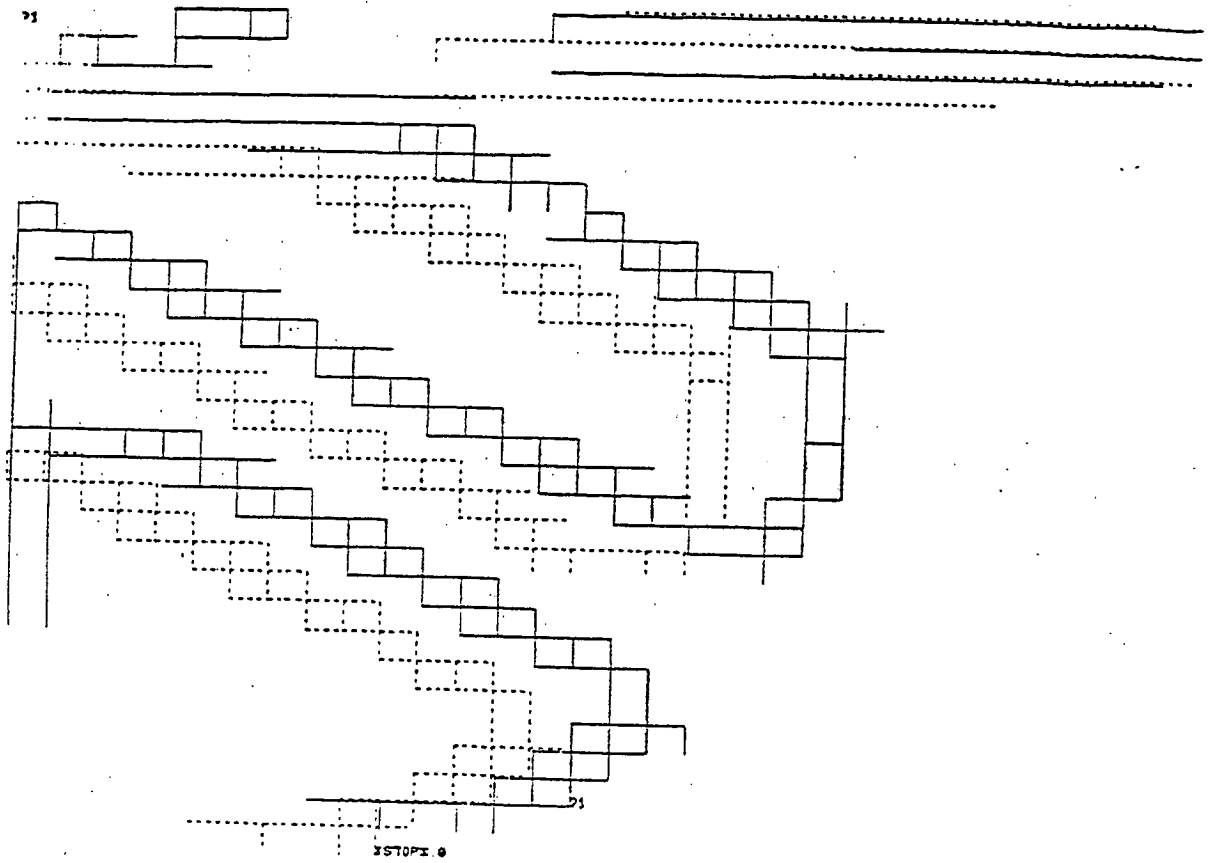


Tab. 3.6



Propagation "SignxSign" avec rupture gain optimal $\gamma=0.150$

a)



b)

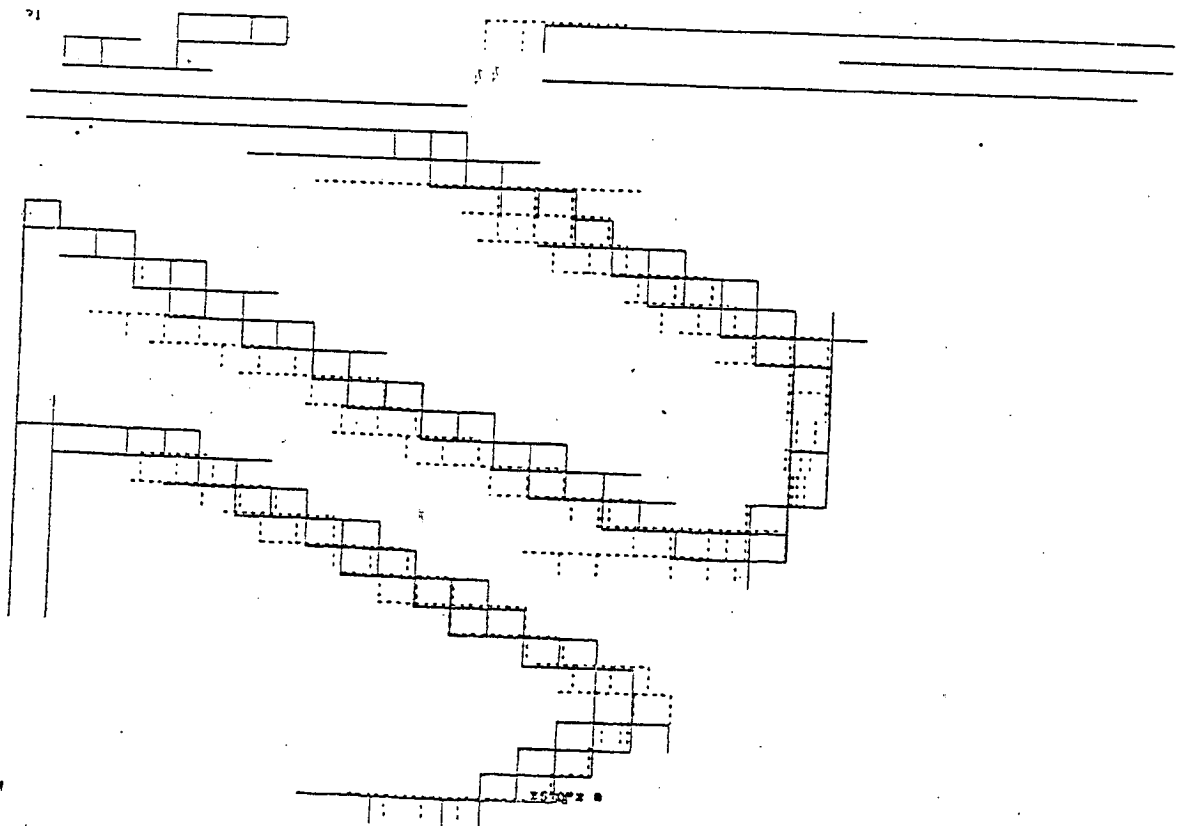


IMAGE NUMERO 1 EN TRAIT PLEIN
IMAGE NUMERO 2 RECALEE EN TIRETE

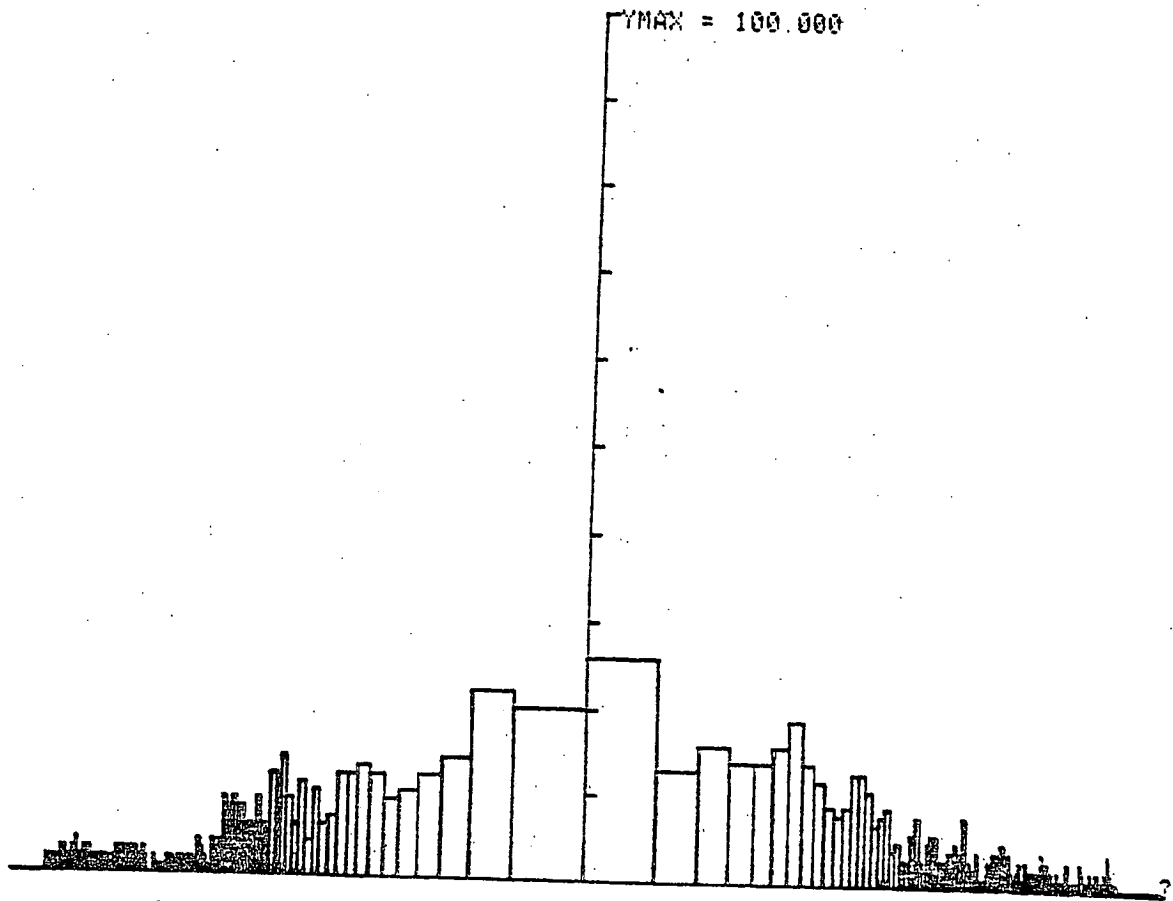
ITEM - 1

ITEM - 2

[illegible]

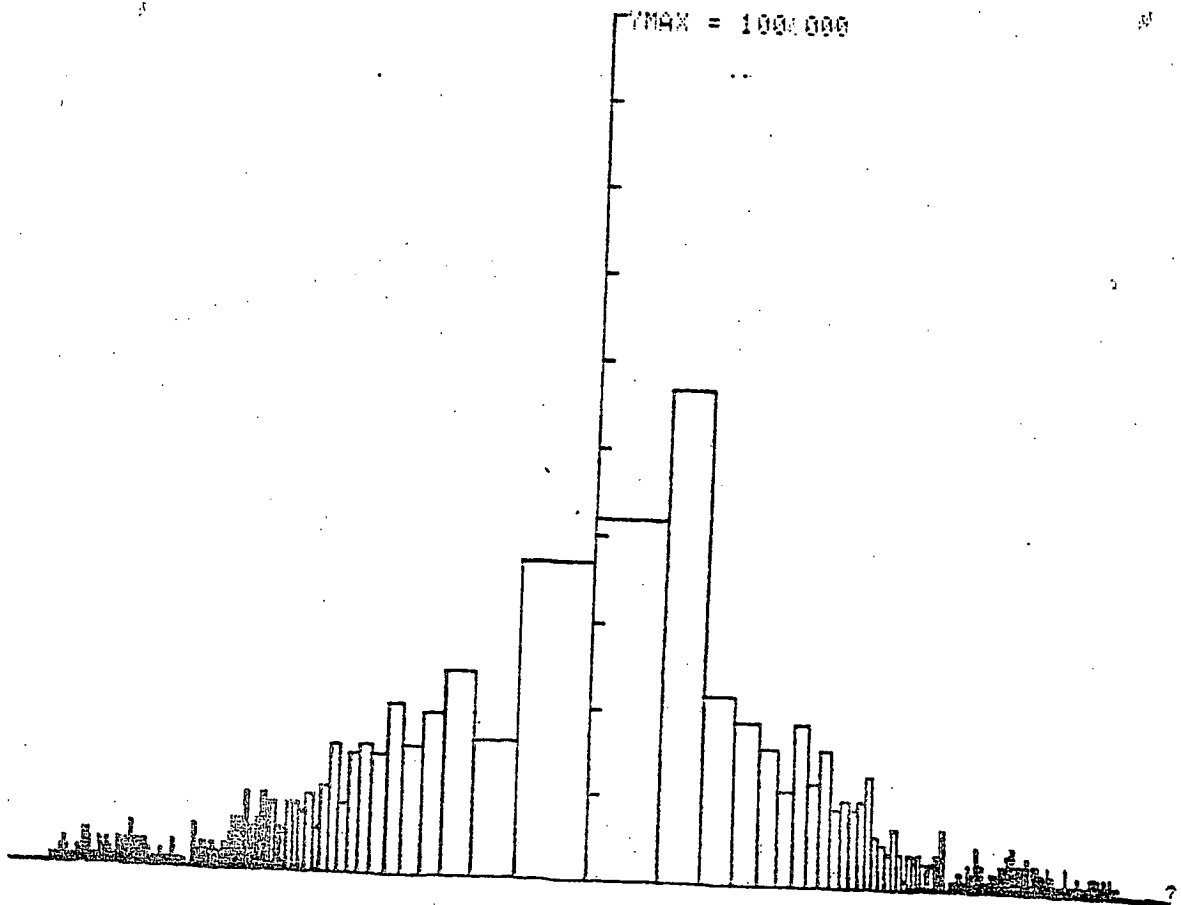
Tab. 3.8

a)

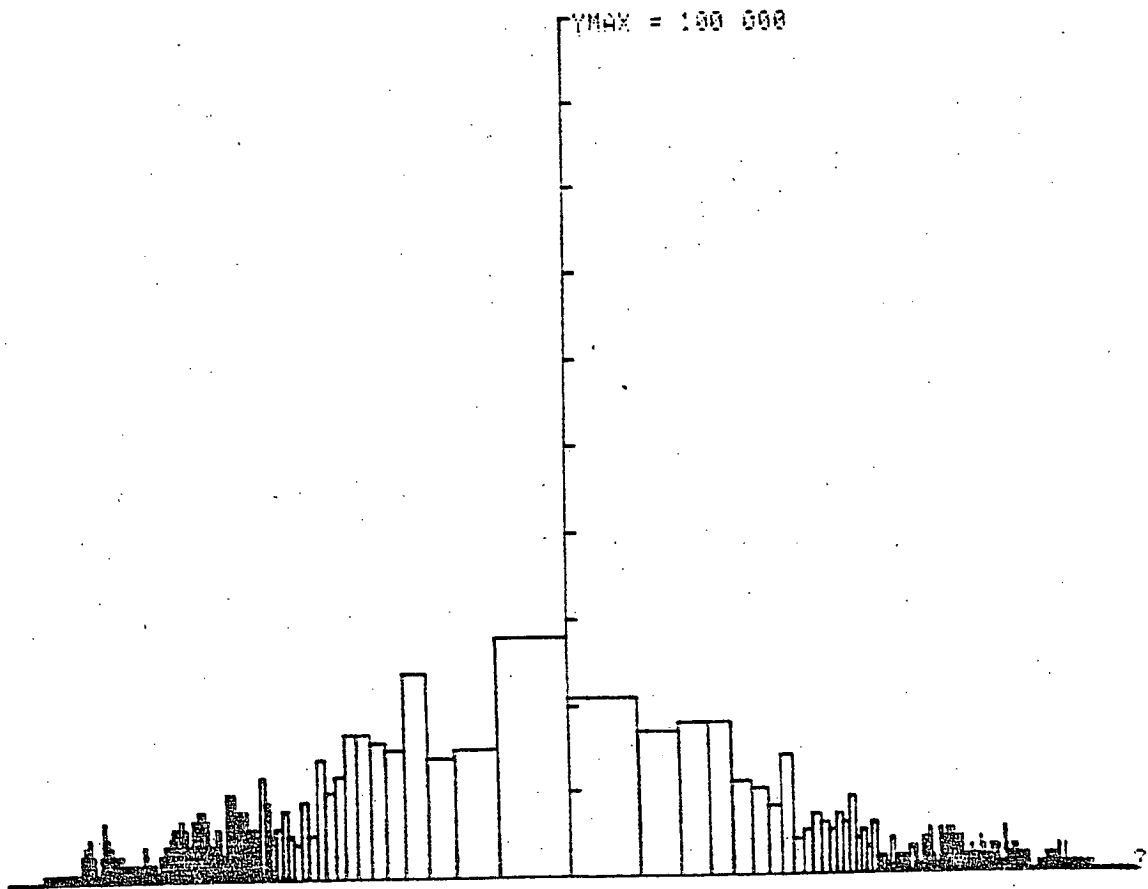


Tab. 3.9

c)

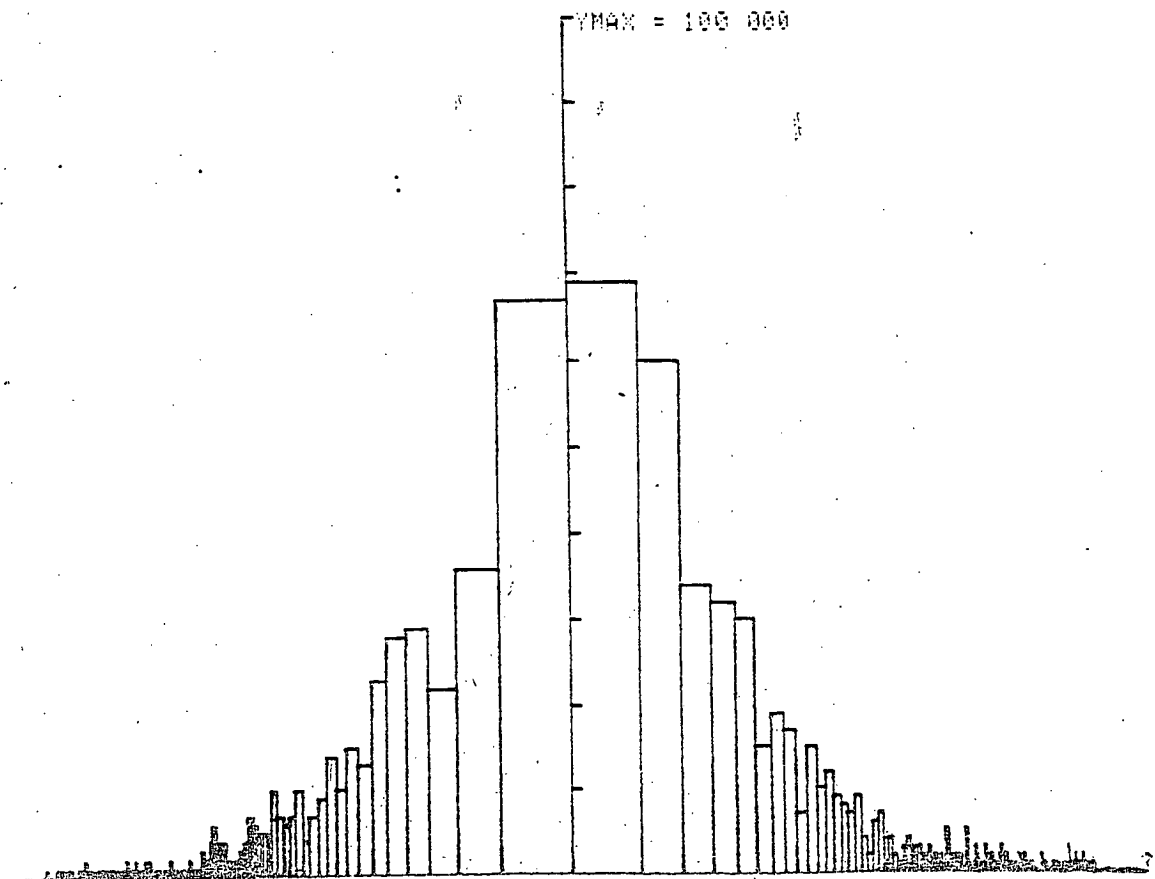


b)

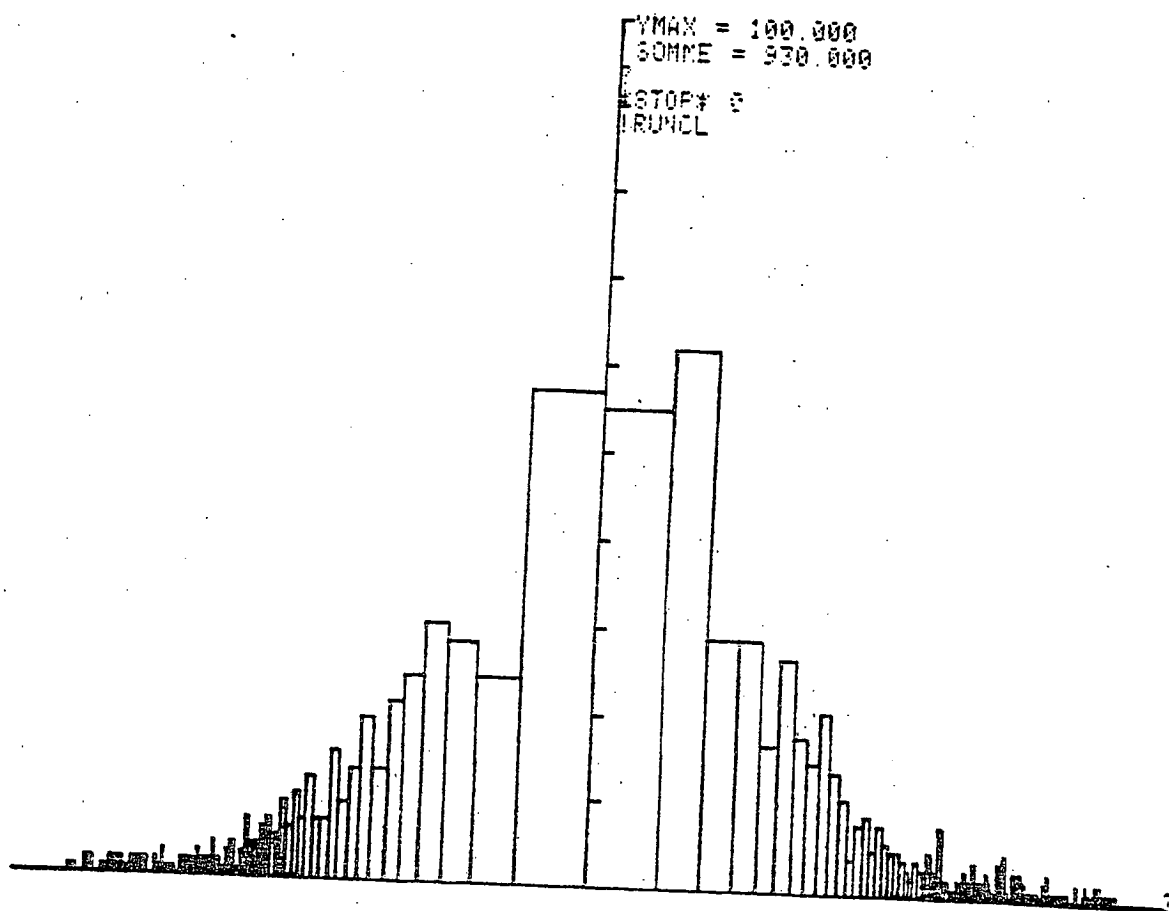


Tab. 3.9

d)

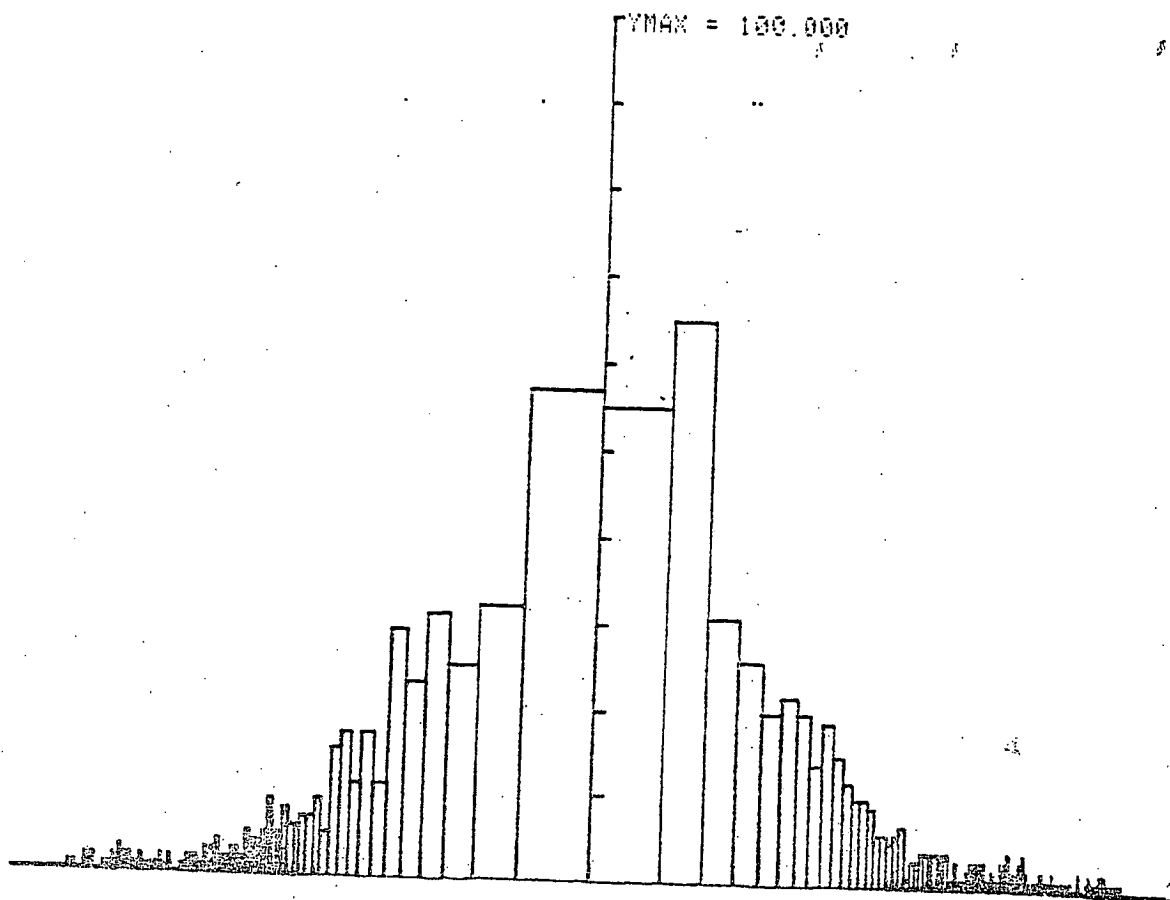


a)

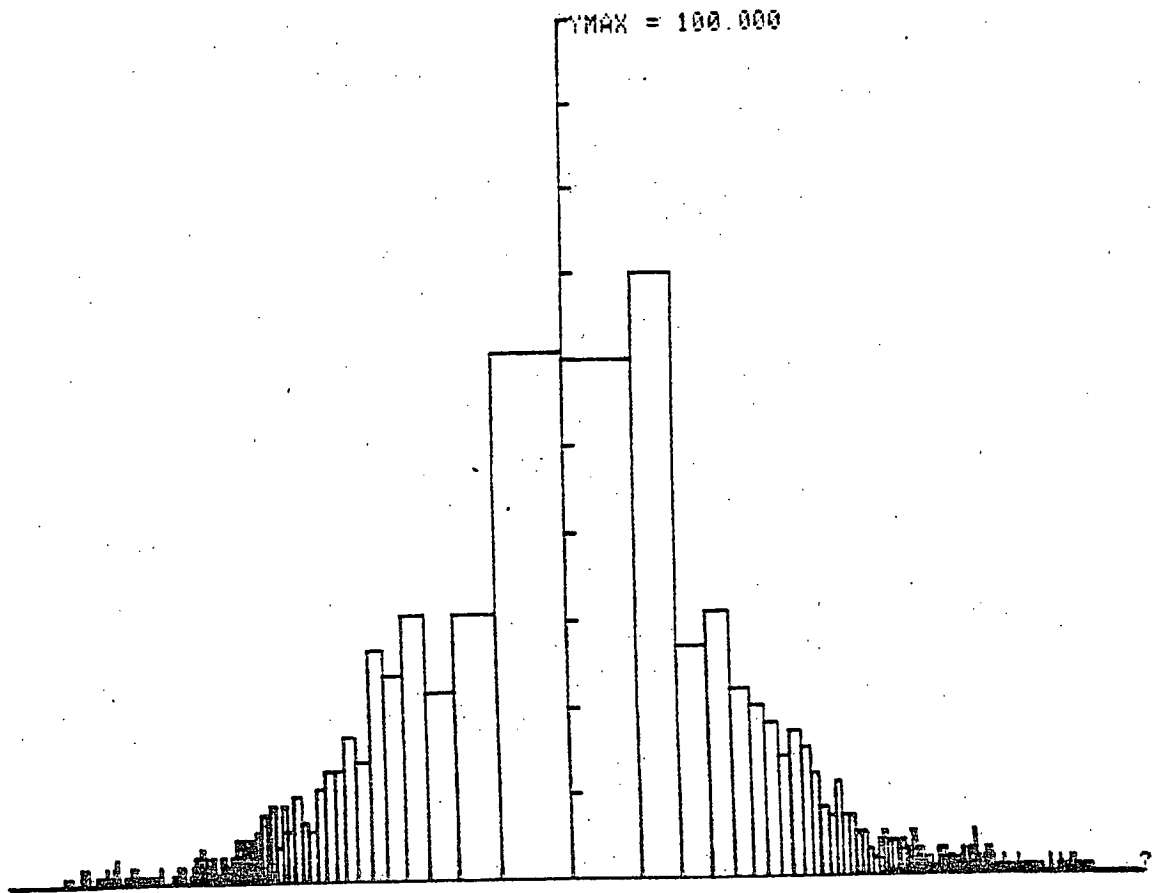


Tab. 3.10

c)

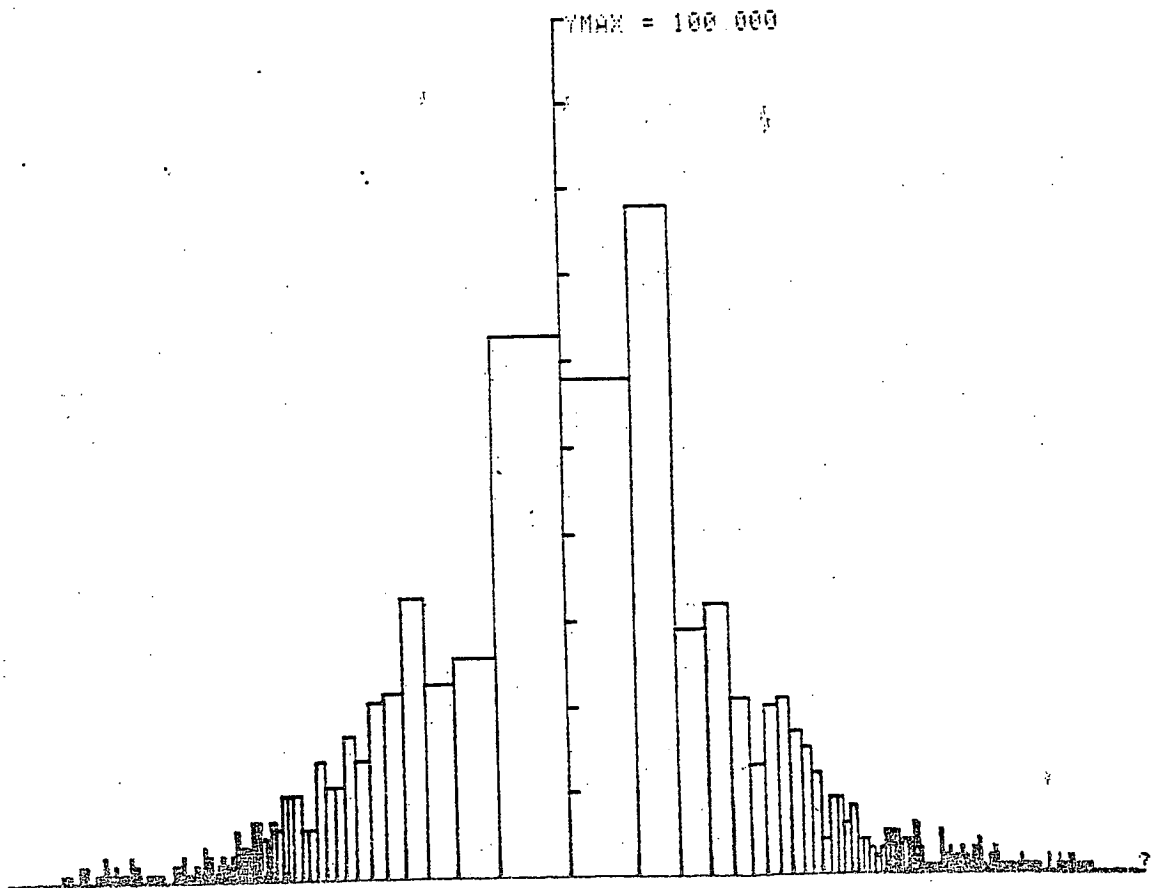


b)

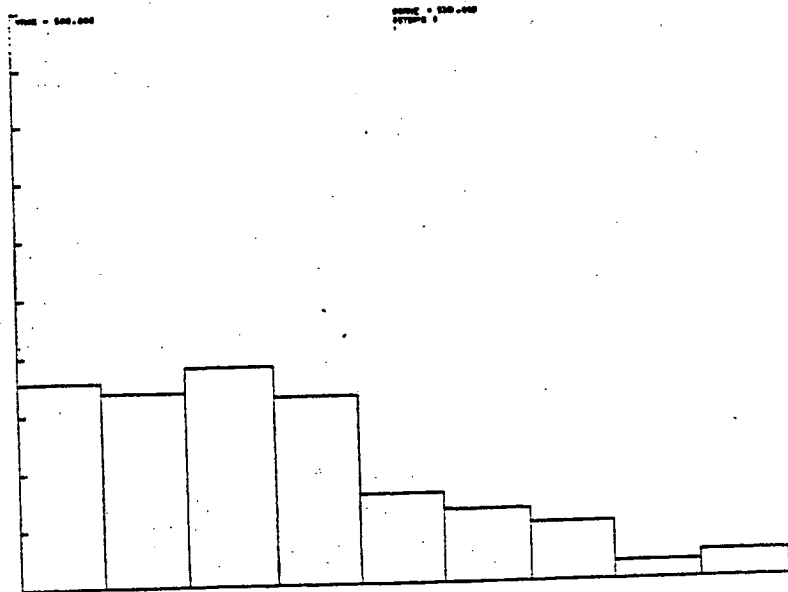


Tab. 3.10

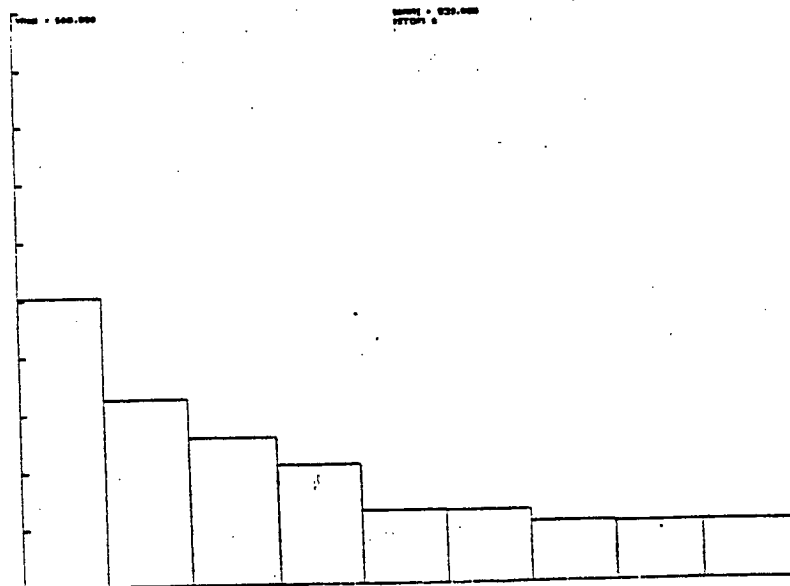
d)



a)

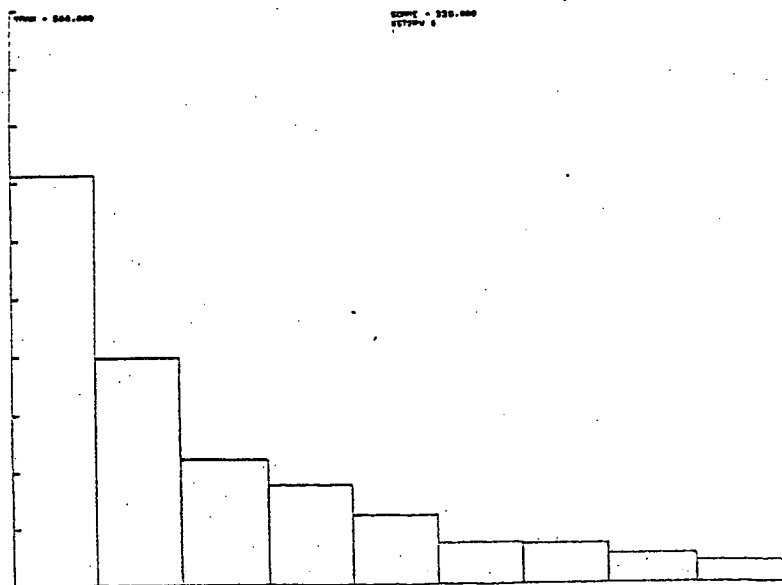


b)

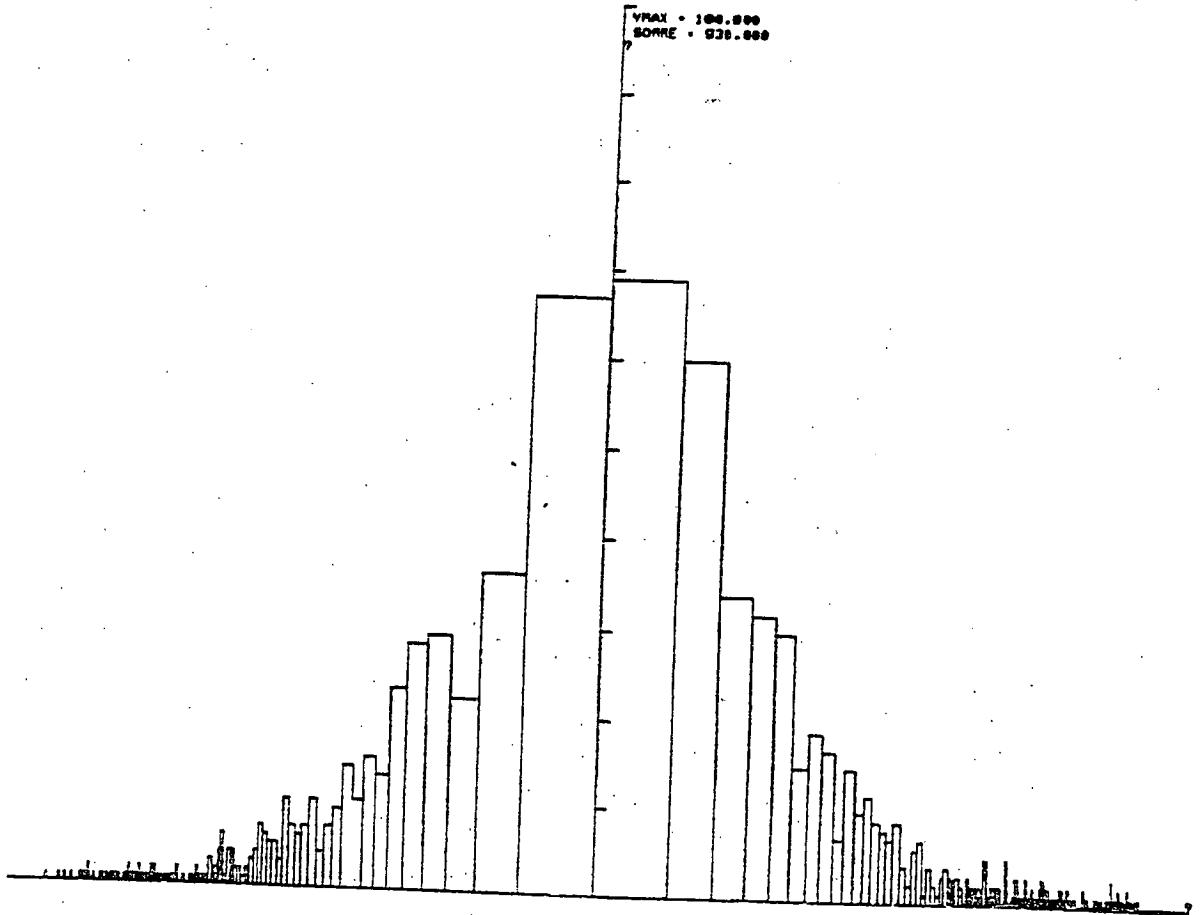


Tab. 3.11

c)

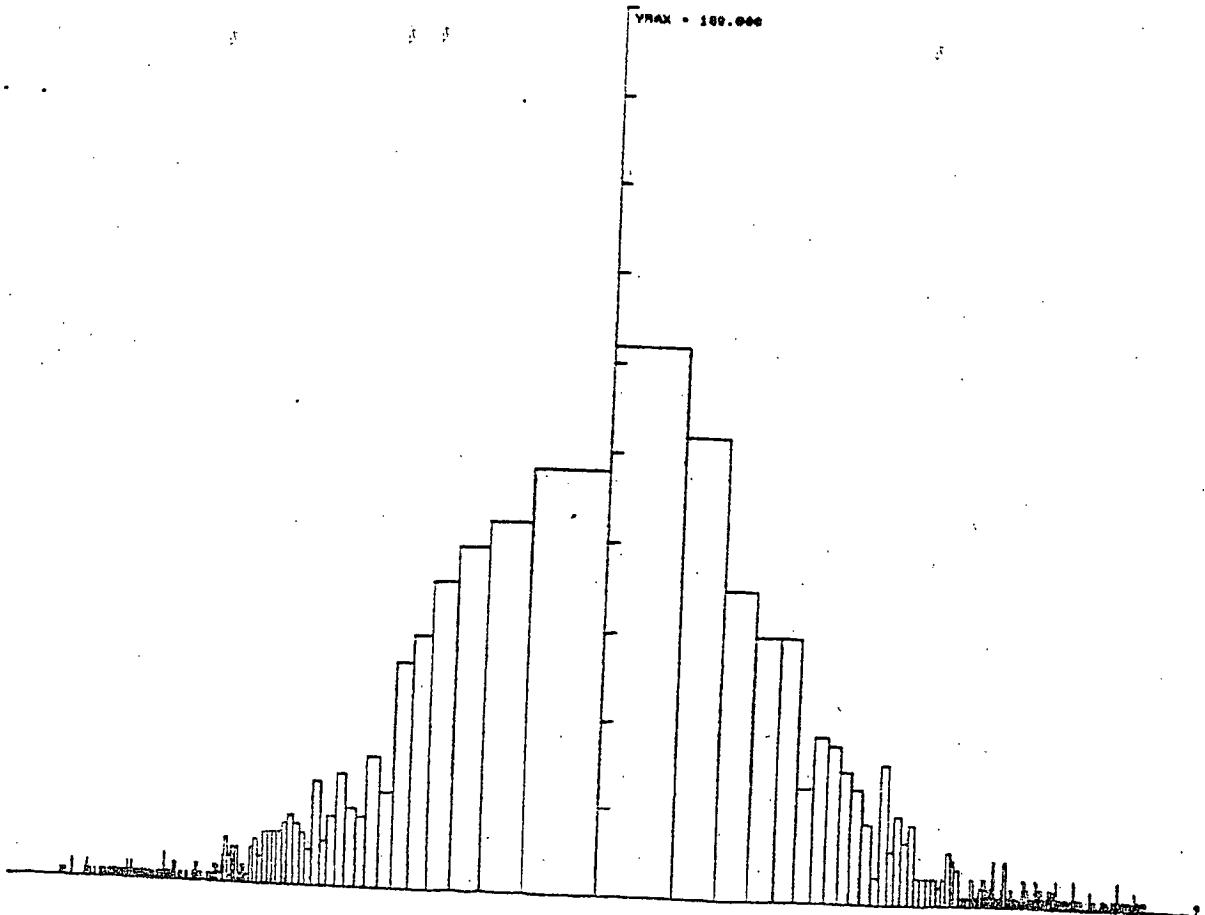


a)

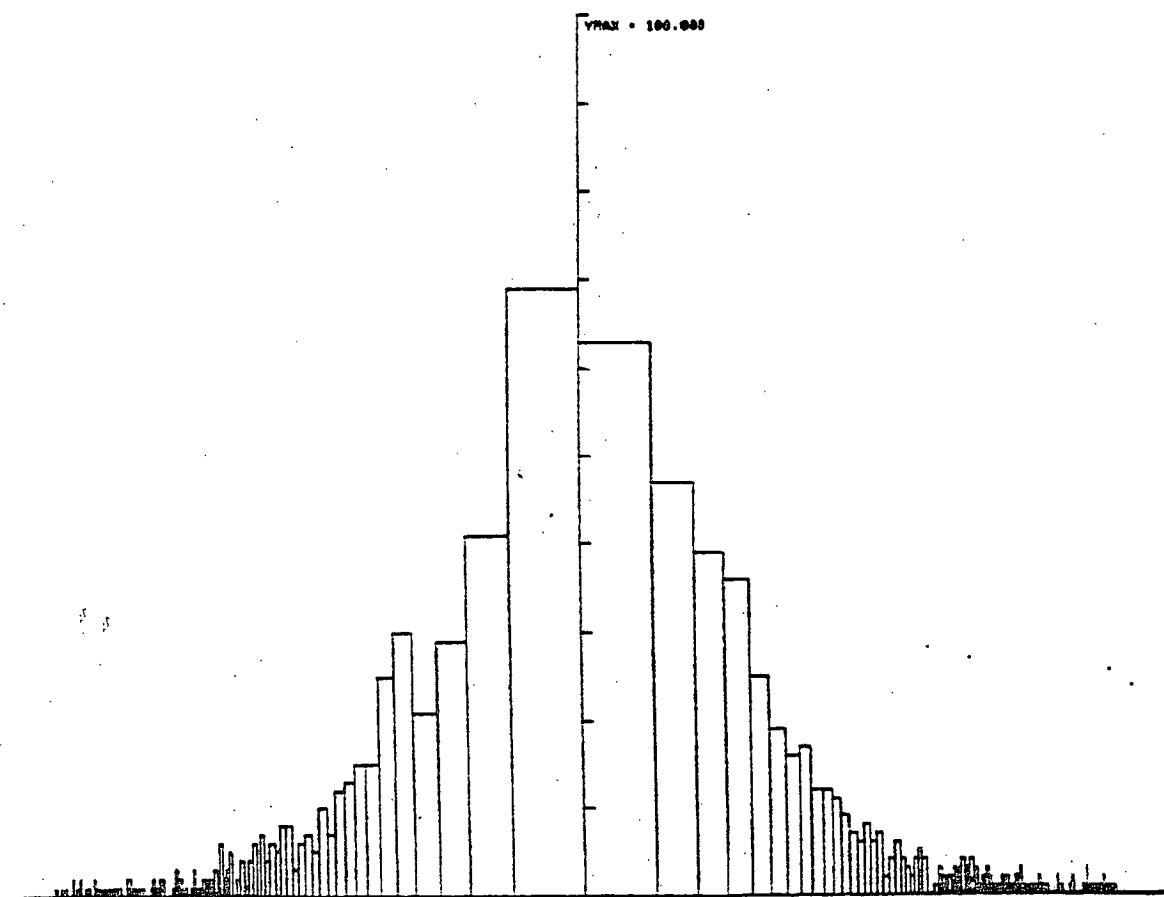


Tab. 3.12

b)



c)



Tab. 3.12

Conclusions :

Pour traiter de façon la meilleure et la plus exhaustive possible l'estimation en tout point du mouvement dans une image de télévision, apparaît comme complément indispensable aux estimations récursives classiques, l'introduction du détecteur de ruptures sur les "frontières" d'objets en mouvement. Les problèmes qu'il a fallu résoudre sont :

- éviter les trop nombreuses ruptures (fausses alarmes) par définition de règles de validation pour les zones-ruptures

- réinitialiser au mieux l'estimateur sur la rupture (s'il y en a une) d'où l'introduction de tests de rupture.

Ceci a permis de résoudre en partie le problème critique des mauvais accrochages des estimateurs récursifs.

Problèmes demeurants :

L'ensemble du mémoire a porté sur une approche très locale (parfois microscopique) de l'estimation du mouvement. La définition et test de l'estimateur ont été faits sur de petites zones d'image. Nous avons donc fait aucunement essayer d'optimiser de manière statistique les divers paramètres (seuils, gain, etc...) introduits.

L'essai (et comparaison) de l'algorithme d'estimation sur images entières de différentes natures (séquences à mouvements divers) serait intéressant. De même la localisation précise des points, source d'erreurs temporelles encore élevées malgré la compensation de mouvement.

Toute cette étude a été menée avec le souci de minimiser la complexité "hardware" pour la construction d'une maquette éventuelle. Nous avons cherché à minimiser :

- le volume d'informations à mémoriser (cf automates)

- la complexité de calculs : additions et multiplications-décalage.

Aucune mesure complète de complexité n'a été faite cependant, notamment en comparaison avec la maquette *PREDITE* réalisant le codeur spatial de Ch. RICHARD.

Comme nous l'avons un peu abordé à propos des problèmes de choix d'interpolations, il serait intéressant de tester à partir de la version complète d'estimation de mouvement telle que nous l'avons définie, des versions simplifiées et regarder les dégradations (dans l'optique codage) causées.

. Il reste également à inclure cet estimateur de mouvement dans un schéma global de prédiction qui pourrait comprendre :

- un codeur intra-image (type *ADPCM*)
- un codeur inter-image avec compensation de mouvement

et une stratégie locale de choix entre ces deux types de codage. Le codage de l'image peut se faire par transmission d'informations d'erreurs de prédiction et d'informations d'adresses. Seul le premier type a été abordé.

Du fait du caractère changeant (par définition !) du mouvement dans une séquence, l'un des problèmes importants est la gestion des mémoires tampons permettant d'observer toute variation brusque du taux de mouvement présent dans la séquence.

BIBLIOGRAPHIE

- [1] : H.H Nagel : Analysis Techniques for Image Sequences
- [2] : P Sallio, F Kretz : CCETT Report CTN/T/13/78 (Nov.78)
Analyse de l'aspect mouvement dans les scènes
télévisuelles en vue de l'élaboration d'un
film pour essais subjectifs
- [3] : J.K Aggarwal, L.S Davis, W.N Martin : Proc IEEE Vol 69 n°5
(mai 81)
Correspondance Processes in
Dynamic Scene Analysis
- [4] : K.H Bers, M Bohrer, H. Gerlach : Object detection in image sequences
S Yalamanchili, W.N Martin, J.K Aggarwal : Differencing operations
for the segmentation of moving objects in dynamic scenes
5th Int. Conf. on Pattern Recognition : (Dec.80) Miami-beach Floride
- [5] : R. Jain, W.N Martin, J.K Aggarwal : Computer graphics and image
processing n°11 (1979)
Segmentation through the detection of changes due to
notion
- [6] : S. Latrous, E. Bourguignat : rapport de stage 81 ENSTB/CCETT
Codage des images de télévision sous échantillonnées
Restitution spatio-temporelle
- [7] : B. Haskell, F.W Mounts, JC Candy : Proc IEEE Vol 60, n°7 (Juil.72)
Interframe Coding of Videotelephone Pictures
- [8] : D.J Connor, J.O Limb : IEEE Trans on Com. n°10 (Oct.74)
Properties of Frame Difference Signals generated by
Moving Images
- [9] : B. Haskell : Advances in Elec. and Elec. Phys supp.12 - P190-217
Frame Replesnishment Coding of T.V.
- [10] : J.O Limb, R.FW Pease, K.A Walsh : BSTJ n°6 (Juil. Aout 74)
Combining Intraframe and Frame-to-Frame coding for T.V
- [11] : J.C Candy, A. Franke, B.G Haskell, F.W Mounts : BSTJ Vol 50 n°6
(juil. Aout 71)
Transmitting Television as clusters of Frame-to-
Frame differences

- [12] : B.G. Haskell, P.L Gordon, K.L Schmidt, J.V Scattaglia : IEEE
trans on Communications vol
n°11 (Nov.77)
Interframe Coding of 525-line monochrome T.V at 1.5Mb/s
- [13] : B. Haskell : IEEE Trans. on Com (Ju 76)
Differential Addressing of Clusters of changed Pictures
Elements for Interframe Coding of Videotelephone signals
- [14] : Brofferio, Cafforio, Del Re, Quaglia, Raccin, Rocca : Alta Frequenza
n°10 (Oct.74)
Redundancy reduction of video signals using mouvement
compensation
- [15] : J.O Limb, J.A. Murphy : Comp graphics and Image Processing n°4 1975
Estimating the velocity of moving Images in T.V images
- [16] : C Cafforio, F Rocca : IEEE Trans Inf. Theory n°5 (sept.76)
Methods for measuring small displacements of T.V images
- [17] : J.O Limb, J.A Murphy : IEEE Trans. on Com. (Avril 78)
Measuring the speed of Moving Objects from T.V signals
- [18] : R. Paquin, E. Dubois : Estimation of the displacement field of mo-
ving images
E. Dubois : A Multimode Interframe Coder using movement compensation
1981 Picture Coding Symposium : summaries of presen-
tations 3-5 Juin81 Montreal
- [19] : R. Paquin : Rapport technique de l'INRS-TELECOM n°80-21.
La détection du mouvement dans les images télévisées (sep
80)
- [20] : A.N Netravali, J.O Robbins : BSTJ Vol 58 n°3 (Mars 79)
Motion-Compensated Television Coding (Part.I)
- [21] : J.O Robbins, A.N Netravali : ICC 79
Interframe Television Coding using movement compensation
- [22] : J.A Stuller, A.N Netravali : BSTJ Vol 58 n°7 (sept.79)
Transform Domain Motion Estimation

- [23] : A.N Netravali, J.A Stuller : BSTJ Vol 58 n°7 (Sept.79)
Motion-Compensated Transform Coding
- [24] : J.A Stuller, A.N Netravali, J.O Robbins : BSTJ 59 n°7 (Sept.80)
Interframe Television Coding using gain and Displacement Compensation
- [25] : mêmes auteurs que 24 : ICC 80
Gain compensated Television Coding
- [26] : A.N Netravali, J.O Robbins : BSTJ 59 n°9 (Nov.80)
Motion-Compensated Coding : some new results
- [27] : K.A Pradhu, A.N Netravali : IEEE on Com. (à paraître)
Motion-compensated color coding : Part I
- [28] : A. Benveniste : RCP CNRS 567 Seminaire Techniques probabilistes en automatique et Telecom. Paris Mars 80
Etude des algorithmes récurrents.
- [29] : A. Benveniste : IEEE Trans on Cont. Vol 27 n°3 (Juin 82)
A measure of the Tracking Capability of Recursive Stochastic Algorithms with Constant gains
- [30] : F. Kretz : Report CCETT/EAS/T/3/81 (Mai 81)
Codage ADPCM en télévision : qualité visuelle, réduction de débit et susceptibilité aux erreurs de transmission
- [31] : F. Kretz, J.L Boudeville, P. Sallio : Revue de radiodiffusion T.V. n°52 (1978)
Qualité subjective en T.V. numérique
- [32] : F. Kretz, J.L Boudeville : Annales des Telecom n°9 - 10 (Sept.Oct.76)
Dégradation des signaux d'images et qualité subjective en codage numérique
- [33] : C. Richard : Thèse 3ème cycle Rennes I. IRISA Report n°124 (Nov.79)
Estimation récursive de l'état local des contours d'image et application à la prédiction adaptative en codage différentiel des signaux de télévision.
- [34] : C. Richard, A. Benveniste, F. Kretz : (à paraître)
Recursive Estimation of local characteristics of edges in T.V. pictures.

[35] : C. Wagner, G. Michel, F. Jorgensen, C. Labit : Convention TDF/CCETT
IRISA n°B8022J

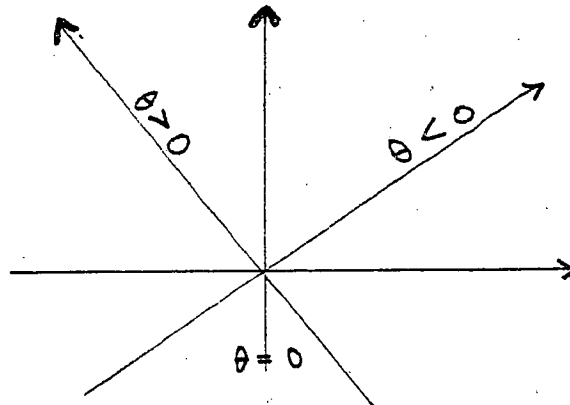
: Predite : Processeur de réduction de débit de transmission des
images T.V. (sept.80)

[36] : J. Sabatier, F. Kretz : Revue UER - Technique n°171 (Oct.78)
L'échantillonnage des composantes du signal de télé-
vision en couleur à 625 lignes.

ANNEXE I

Discrétisation des orientations de contours.

* la convention de signe pour les contours est la suivante :



* 15 valeurs quantifiées distinctes ont été choisies pour l'estimateur d'orientation $\hat{\theta} = \hat{H}/\hat{V}$. Les seuils et valeurs de quantification sont résumées dans le tableau A.1 suivant :

$Z = H/V$	θ
$ Z \geq 7.6$	8
$4.6 \leq Z < 7.6$	7
$2.6 \leq Z < 4.6$	6
$1.4 \leq Z < 2.6$	5
$0.9 \leq Z < 1.4$	4
$0.4 \leq Z < 0.9$	3
$0.2 \leq Z < 0.4$	2
$ Z < 0.2$	1
contour vertical	0

ANNEXE III

DOCUMENTS PHOTOGRAPHIQUES

- Φ 1 : Images de la séquence "Judith"
(Cf. résultats de traitement p.48, 60 à 62)
Sont visualisées 4 images consécutives extraites respectivement
en a) : du début de la séquence
en b) : au milieu de la séquence
en c) : en fin de séquence
- Φ 2 : Images de la séquence "Chateau"
a) début de séquence
b) milieu de séquence
c) fin de séquence
- Φ 3 : Images binaires de gradients sur "Chateau"
a) gradients verticaux détectés sur image t (t = 41)
b) gradients validés (Cf p 53) de l'image précédente(t - 1)
(Horiz. et Vert.)
- Φ 4 : Images différences temporelles (sur "Chateau"
a) Différences interimages : Image 41 - Image 40
b) Image négative de la précédente avec quantification.
non uniforme
c) Différences interimages : Image 41 - Image 25
- Φ 5 : Comparaison d'images de "DFD" : (Displaced Frame Difference)
$$DFD(x,t) = I(x,t) - I(x-D, t-\tau)$$

(t = 41)
a) Différences interimages
b) Estimation de mouvement sans rupture et avec propagation
VI x DFD : PSRO (Cf p 87)
c) Estimation de mouvement sans rupture, avec propagation
"simplifiée" PSR1 (Cf p 88) et troncature du mouvement
à 5 pel/image.
d) Algorithme final avec rupture, propagation PSR1 et
troncature à 5 pel/image.

Ces images ont été tirées avec les niveaux de contraste et luminosité standard du moniteur TV (à comparer avec les photos ϕ_1 et ϕ_2)

ϕ_6 : Même résultats qu'à la planche ϕ_5 :
Les niveaux de contraste et luminosité ont été augmentés afin de visualiser les divers niveaux d'erreurs.

ANNEXE II

Définition de l'état interpolé $\tilde{\mathbf{E}}$ d'une fourche spatiale

Au numéro de marche (k), les deux automates d'états $\hat{\mathbf{E}}^1_{k-1}$ et $\hat{\mathbf{E}}^2_{k-1}$ fusionnent pour donner l'état interpolé $\hat{\mathbf{E}}_{k-1}$ défini par le tableau ci-dessous A.2

Calcul de l'état interpolé dans le cadre des fourches spatiales

$\hat{\mathbf{G}}^1_{k-1}$	$\hat{\mathbf{G}}^2_{k-1}$	$\tilde{\mathbf{G}}_{k-1} = (\tilde{\mathbf{N}}_{k-1}, \tilde{\mathbf{H}}_{k-1}, \tilde{\mathbf{V}}_{k-1}, \tilde{\mathbf{F}}_{k-1}, \tilde{\mathbf{Q}}_{k-1})$
$\hat{\mathbf{N}}^1_{k-1}$	$\hat{\mathbf{N}}^2_{k-1}$	$\tilde{\mathbf{N}}_{k-1} = \max(\hat{\mathbf{N}}^1_{k-1}, \hat{\mathbf{N}}^2_{k-1})$
$\hat{\mathbf{H}}^1_{k-1}$	$\hat{\mathbf{H}}^2_{k-1}$	$\tilde{\mathbf{H}}_{k-1} = \alpha \cdot \hat{\mathbf{H}}^1_{k-1} \cdot \hat{\mathbf{S}}^1_{k-1} + (1-\alpha) \cdot \hat{\mathbf{H}}^2_{k-1} \cdot \hat{\mathbf{S}}^2_{k-1}$
$\hat{\mathbf{V}}^1_{k-1}$	$\hat{\mathbf{V}}^2_{k-1}$	$\tilde{\mathbf{V}}_{k-1} = \alpha \cdot \hat{\mathbf{V}}^1_{k-1} + (1-\alpha) \cdot \hat{\mathbf{V}}^2_{k-1}$
$\hat{\mathbf{F}}^1_{k-1}$	$\hat{\mathbf{F}}^2_{k-1}$	$\tilde{\mathbf{F}}_{k-1} = \alpha \cdot \hat{\mathbf{F}}^1_{k-1} + (1-\alpha) \cdot \hat{\mathbf{F}}^2_{k-1}$
$\hat{\mathbf{Q}}^1_{k-1}$	$\hat{\mathbf{Q}}^2_{k-1}$	$\tilde{\mathbf{Q}}_{k-1} = 0.$

Le facteur de pondération α est défini de manière à privilégier l'automate de plus grande ancienneté.

$\hat{\mathbf{N}}^1_{k-1} = \hat{\mathbf{N}}^2_{k-1}$	$\alpha = 1/2$
$\hat{\mathbf{N}}^1_{k-1} > \hat{\mathbf{N}}^2_{k-1}$	$\alpha = 3/4$
$\hat{\mathbf{N}}^1_{k-1} < \hat{\mathbf{N}}^2_{k-1}$	$\alpha = 1/4$
$\hat{\mathbf{N}}^1_{k-1} = 1$	$\alpha = 0$
$\hat{\mathbf{N}}^2_{k-1} = 1$	$\alpha = 1$

Liste des Publications Internes IRISA

- PI 156 **Réursion induction principe revisited**
G. Boudol, L. Kott, 49 pages ; Décembre 1981
- PI 157 **Loi d'une variable aléatoire à valeur R^* réalisant le minimum des moments d'ordre supérieur à deux lorsque les deux premiers sont fixés**
M. Kowalowka, R. Marie, 8 pages ; Décembre 1981
- PI 158 **Réalisations stochastiques de signaux non stationnaires, et identification sur un seul échantillon**
A. Benveniste J.J. Fuchs, 33 pages ; Mars 1982
- PI 159 **Méthode d'interprétation d'une classification hiérarchique d'attributs-modalités pour l'«explication» d'une variable ; application à la recherche de seuil critique de la tension artérielle systolique et des indicateurs de risque cardiovasculaire**
B. Tallur, 34 pages ; Janvier 1982
- PI 160 **Probabilité stationnaire d'un réseau de files d'attente multiclasse à serveur central et à routages dépendant de l'état**
L.M. Le Ny, 18 pages ; Janvier 1982
- PI 161 **Détection séquentielle de changements brusques des caractéristiques spectrales d'un signal numérique**
M. Basseville, A. Benveniste, pages ; Mars 1982
- PI 162 **Actes regroupés des journées de Classification de Toulouse (Mai 1980), et de Nancy (Juin 1981)**
I.C. Lerman, 304 pages ;
- PI 163 **Modélisation et Identification des caractéristiques d'une structure vibratoire : un problème de réalisation stochastique d'un grand système non stationnaire**
M. Prevosto, A. Benveniste, B. Barnouin, 46 pages ; Mars 1982
- PI 164 **An enlarged definition and complete axiomatization of observational congruence of finite processes**
Ph. Darondeau, 45 pages ; Avril 1982
- PI 165 **Accès vidéotex à une banque de données médicales**
A. Chauffaut, M. Dragone, R. Rivoire, J.M. Roger, 25 pages ; Mai 1982
- PI 166 **Comparaison de groupes de variables définies sur le même ensemble d'individus**
B. Escofier, J. Pages, 115 pages ; Mai 1982
- PI 167 **Transport en circuits virtuels internes sur réseau local et connexion Transpac**
M. Tournois, R. Trépos, 90 pages ; Mai 1982
- PI 168 **Impact de l'intégration sur le traitement automatique de la parole.**
P. Quinton, 14 pages ; Mai 1982
- PI 169 **A systolic algorithm for connected word recognition**
J.P. Banâtre, P. Frison, P. Quinton, 13 pages ; Mai 1982
- PI 170 **A network for the detection of words in continuous speech**
J.P. Banâtre, P. Frison, P. Quinton, 24 pages ; Mai 1982
- PI 171 **Le langage ADA : Etude bibliographique**
J. André, Y. Jégou, M. Raynal, 12 pages ; Juin 1982
- PI 172 **Comparaison de groupes de variables : 2ème partie : un exemple d'application**
B. Escofier, J. Pajès, 37 pages ; Juillet 1982
- PI 173 **Unfold-fold program transformations**
L. Kott, 29 pages ; Juillet 1982
- PI 174 **Remarques sur les langages de parenthèses**
J.M. Autebert, J. Beauquier, L. Boasson, G. Senizergues, 20 pages ; Juillet 1982
- PI 175 **Langages de parenthèses, langages N.T.S. et homomorphismes inverses**
J.M. Autebert, L. Boasson, G. Senizergues, 26 pages ; Juillet 1982
- PI 176 **Tris pour machines synchrones ou Baudet Stevenson revisited**
R. Rannou, 26 pages ; Juillet 1982
- PI 177 **Un nouvel algorithme de classification hiérarchique des éléments constitutifs de tableau de contingence basé sur la corrélation**
B. Tallur, Juillet 1982 ;
- PI 178 **Programmes d'analyse des résultats d'une classification automatique**
I.C. Lerman et collaborateurs, 79 pages ; Septembre 1982
- PI 179 **Attitude à l'égard des mathématiques des élèves de sixième**
J. Degouys, R. Gras, M. Postic, 29 pages ; Septembre 1982
- PI 180 **Traitements de textes et manipulations de documents : bibliographie analytique.**
J. André, 20 pages ; Septembre 1982
- PI 181 **Algorithme assurant l'insertion dynamique d'un processeur autour d'un réseau à diffusion et garantissant la cohérence d'un système de numérotation des paquets global et réparti**
Annick Le Coz, Hervé Le Goff, Michel Ollivier, 31 pages ; Octobre 1982
- PI 182 **Interprétation non linéaire d'un coefficient d'association entre modalités d'une juxtaposition de tables de contingence**
Israël César Lerman, 34 pages ; Novembre 1982
- PI 183 **L'IRISA vu à travers les stages effectués par ses étudiants de DEA (1ère année de thèse)**
Daniel Herman, 41 pages ; Novembre 1982
- PI 184 **Commande non linéaire robuste des robots manipulateurs**
Claude Samson, 52 pages ; Janvier 1983
- PI 185 **Dialogue et représentation des informations dans un système de messagerie intelligent**
Philippe Besnard, René Quiniou, Patrice Quinton, Patrick Saint-Dizier, Jacques Siroux, Laurent Trilling, 45 pages ; Janvier 1983
- PI 186 **Analyse classificatoire d'une correspondance multiple ; typologie et régression**
I.C. Lerman, 54 pages ; Janvier 1983
- PI 187 **Estimation de mouvement dans une séquence d'images de télévision en vue d'un codage avec compensation de mouvement**
Claude Labit, 132 pages ; Janvier 1983

